



**HAL**  
open science

# Zones d'écoute personnalisées mobiles par approches adaptatives

Guilhem Pages

► **To cite this version:**

Guilhem Pages. Zones d'écoute personnalisées mobiles par approches adaptatives. Acoustique [physics.class-ph]. Le Mans Université, 2024. Français. NNT : 2024LEMA1012 . tel-04696545

**HAL Id: tel-04696545**

**<https://theses.hal.science/tel-04696545v1>**

Submitted on 13 Sep 2024

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

# THÈSE DE DOCTORAT

DE

**LE MANS UNIVERSITÉ**

SOUS LE SCEAU DE

LA COMUE ANGERS – LE MANS

ÉCOLE DOCTORALE N° 602

*Sciences de l'Ingénierie et des Systèmes*

Spécialité : *Acoustique*

Par

**Guilhem PAGÈS**

**Sound zones mobiles**

« Zones d'écoute personnalisées mobiles par approches adaptatives »

**Thèse présentée et soutenue à Le Mans, le 23 août 2024**

**Unité de recherche : LAUM, UMR CNRS 6613**

**Thèse N° : 2024LEMA1012**

## **Rapporteurs avant soutenance :**

Philippe-Aubert GAUTHIER Professeur, École des arts visuels et médiatiques, Montréal, Canada  
Daniel MASSICOTTE Professeur, Université du Québec, Trois-Rivières, Canada

## **Composition du Jury :**

Président :	Thomas HÉLIE	Directeur de recherche, IRCAM, Paris, France
Examineurs :	Philippe-Aubert GAUTHIER	Professeur, École des arts visuels et médiatiques, Montréal, Canada
	Hervé LISSEK	Maître d'Enseignement et de Recherche, Lausanne, Suisse
	Daniel MASSICOTTE	Professeur, Université du Québec, Trois-Rivières, Canada
Dir. de thèse :	Manuel MELON	Professeur des Universités, LAUM, Le Mans, France
Co-dir. de thèse :	Laurent SIMON	Professeur des Universités, LAUM, Le Mans, France
Co-dir. de thèse :	Roberto LONGO	Chercheur sénior, IMEC, Leuven, Belgique



*À mon grand-père Jean-Louis, qui rêvait de faire des études*





# REMERCIEMENTS

---

Je tiens tout d'abord à exprimer ma profonde gratitude à ma direction de thèse, Manuel, Laurent et Roberto, pour leur soutien indéfectible, leurs conseils éclairés et leur patience tout au long de cette aventure. Leur expertise et leur engagement ont été essentiels à la réalisation de ce travail.

Je remercie également les membres de mon jury, pour avoir accepté d'évaluer ce travail et pour leurs précieuses observations qui ont permis d'enrichir ce document.

Mes remerciements vont également à l'ensemble des enseignants, professeurs, chercheurs, personnels administratifs et techniques du Laboratoire d'Acoustique de l'Université du Mans, dont les enseignements et les discussions ont nourri et inspiré mes réflexions.

Je tiens également à remercier Valentin et Luna, étudiants à Le Mans Université, de m'avoir supporté en tuteur de projet deux années consécutives, et d'avoir participé aux expériences sur les *sound zones* ainsi que leurs apports sur le choix de la géométrie optimale.

Je n'oublie pas les doctorants, post-docs et étudiants du laboratoire, pour la bonne humeur et les moments conviviaux, la cuisine et les BBQ, les week-ends et les soirées.

Enfin, je remercie mes parents, ma famille et mes amis qui ont su m'aider et m'accompagner dans ce projet.



# SOMMAIRE

---

Liste des figures	XI
Liste des tableaux	XIII
Notations	XV
<b>Introduction</b>	<b>1</b>
Contexte et introduction aux <i>sound zones</i> . . . . .	1
Organisation du document et notations . . . . .	3
Liens internet . . . . .	3
<b>I Estimation de réponses impulsionnelles</b>	<b>7</b>
<b>1 Position du problème</b>	<b>8</b>
1.1 Sound zones et réponses impulsionnelles . . . . .	9
1.2 Réduction du système MIMO à M systèmes MISO . . . . .	9
<b>2 État de l’art</b>	<b>11</b>
2.1 <i>Exponential swept sine</i> pour système SISO . . . . .	12
2.2 Signaux orthogonaux pour systèmes MISO . . . . .	13
2.3 Méthodes adaptatives en contexte SISO . . . . .	14
2.3.1 Filtrage de Wiener en contexte stationnaire . . . . .	17
2.3.2 Algorithme <i>Recursive Least Square</i> (RLS) . . . . .	18
2.3.3 Algorithme <i>Least Mean Square</i> (LMS) . . . . .	20
2.3.4 Algorithme NLMS . . . . .	22
2.3.5 Algorithme VSS-LMS . . . . .	23
2.3.6 Algorithme de méta-gradient . . . . .	25
2.4 Extension de SISO à MISO . . . . .	26
2.5 Conclusion . . . . .	29

<b>3</b>	<b>Algorithme MISO-Autostep</b>	<b>30</b>
3.1	Développement théorique . . . . .	30
3.2	Métriques et optimisation préliminaire . . . . .	36
3.2.1	Métriques . . . . .	36
3.2.2	Choix des méta-paramètres . . . . .	37
3.2.3	Temps de calcul . . . . .	40
3.3	Comparaison des algorithmes MISO-LMS et MISO-Autostep dans un environnement simulé . . . . .	41
3.4	Validation expérimentale . . . . .	45
3.4.1	Dispositif expérimental . . . . .	46
3.4.2	Mesures en salle semi-anéchoïque . . . . .	46
3.4.3	Mesures dans une salle du laboratoire . . . . .	48
3.5	Conclusion . . . . .	52
<b>4</b>	<b>Extension aux systèmes variant dans le temps</b>	<b>53</b>
4.1	Systèmes linéaires variant dans le temps (LTV) . . . . .	53
4.2	Test de l'algorithme MISO-AS dans un cas LTV . . . . .	55
<b>5</b>	<b>Conclusion</b>	<b>56</b>
<b>II</b>	<b>Sound zones</b>	<b>57</b>
<b>6</b>	<b>Contexte &amp; état de l'art</b>	<b>58</b>
6.1	Introduction . . . . .	58
6.2	Etat de l'art des méthodes fonctionnant dans le domaine temporel . . . . .	61
6.3	Sound zones dynamiques . . . . .	63
6.4	Formulation mathématique du problème . . . . .	64
6.4.1	Cas 1 : Sans prendre en compte le signal d'entrée . . . . .	65
6.4.2	Cas 2 : En prenant en compte le signal d'entrée . . . . .	67
6.5	Algorithme BACC . . . . .	69
6.6	Algorithme BACC-PM . . . . .	70
6.7	Conclusion . . . . .	72
<b>7</b>	<b>Version récursive de l'algorithme BACC-PM</b>	<b>73</b>
7.1	Résolution par descente de gradient . . . . .	73

---

7.1.1	Étude de convergence . . . . .	75
7.1.2	Gradient conjugué . . . . .	76
7.2	Simulations en contexte statique . . . . .	79
7.2.1	Géométrie . . . . .	79
7.2.2	Métriques . . . . .	80
7.2.3	Résultats . . . . .	82
7.3	Conclusion . . . . .	90
<b>8</b>	<b>Zones en mouvement et réponses impulsionnelles connues</b>	<b>91</b>
8.1	Inversion de zones . . . . .	91
8.2	Déplacement continu . . . . .	100
8.2.1	Condition initiale nulle . . . . .	101
8.2.2	Condition initiale fournie par BACC-PM . . . . .	101
8.2.3	Réponses impulsionnelles connues tous les 10 cm . . . . .	103
8.3	Conclusion . . . . .	105
<b>9</b>	<b>Zones en mouvement et réponses impulsionnelles inconnues : Perspectives</b>	<b>107</b>
9.1	Algorithmes MISO-Autostep et BACC-PM-Gd (boucle ouverte) . . . . .	109
9.2	Algorithmes MISO-Autostep et BACC-PM-FxLMS (boucle fermée) . . . . .	110
	<b>Conclusion</b>	<b>117</b>
	<b>Annexes</b>	<b>122</b>
<b>A</b>	<b>Propriétés de convergence de l’algorithme MISO-Autostep</b>	<b>122</b>
<b>B</b>	<b>Choix de la géométrie</b>	<b>125</b>
B.1	Sélection des sources . . . . .	125
B.2	Sélection des microphones . . . . .	126
	<b>Bibliographie</b>	<b>129</b>



# TABLE DES FIGURES

---

1.1	Schéma de principe d'un système audio de <i>sound zones</i> . . . . .	8
2.1	Schéma bloc d'un filtre adaptatif pour l'estimation de réponses impulsionnelles. En rouge les signaux connus, en jaune la réponse impulsionnelle à estimer et en vert le filtre que l'algorithme adaptatif peut modifier. . . . .	16
3.1	Schéma bloc du système MISO. . . . .	31
3.2	Fonction de coût $J(\kappa, \gamma)$ en fonction des méta-paramètres $\kappa$ et $\gamma$ . Les points bleus représentent les calculs de $J$ tandis que la surface orange représente le modèle de régression du processus gaussien. Pour une large plage de valeurs de $(\kappa, \gamma)$ , $J(\kappa, \gamma) \simeq 5,1 \times 10^{-6}$ . La valeur minimale de $J(\kappa, \gamma)$ est obtenue pour $(\kappa, \gamma) = (0,22, 4,7 \times 10^{-10})$ . . . . .	38
3.3	<i>Learning curves</i> de l'algorithme MISO-Autostep pour différentes valeurs de $\gamma$ avec $\kappa = 0.01$ . . . . .	39
3.4	<i>Learning curves</i> de l'algorithme MISO-Autostep pour différentes valeurs de $\kappa$ avec $\gamma = 0.01$ . . . . .	40
3.5	<i>Learning curves</i> des algorithmes MISO-Autostep et MISO-LMS. Pour MISO-LMS, trois valeurs de $\mu$ sont présentées. Pour MISO-Autostep (MISO-AS), $\kappa = 0.01$ et $\gamma = 0.01$ . . . . .	40
3.6	Géométrie simulée. Les six haut-parleurs (noir) sont espacés de 50 cm, le microphone apparaît en gris clair. . . . .	42
3.7	<i>Learning curves</i> avec $K = 100$ dans (3.20). Gris : MISO-LMS avec $\mu^{(l)} = 1/(NL + 1)$ . Noir : MISO-Autostep avec $\kappa = 0.01, \gamma = 10$ . . . . .	42
3.8	Réponse impulsionnelle (RI) du haut-parleur 1 dans la salle simulée : (a) après 5 s, (b) après 60 s. La vraie RI est en gris, alors que la RI estimée avec MISO-Autostep est en noir ( $\kappa = 0.01, \gamma = 10$ ). L'erreur $\epsilon$ à la fin de la simulation est en bas des graphiques. . . . .	43



3.9	Réponse impulsionnelle (RI) du haut-parleur 6 dans la salle simulée : (a) après 5 s, (b) après 60 s. La vraie RI est en gris, alors que la RI estimée avec MISO-Autostep est en noir ( $\kappa = 0.01$ , $\gamma = 10$ ). L'erreur $\epsilon$ à la fin de la simulation est en bas des graphiques. . . . .	44
3.10	FRF (module) du haut-parleur 1 dans la salle simulée. Gris : vraie FRF. Noir : estimée avec Autostep après 60 s. . . . .	45
3.11	FRF (module) du haut-parleur 6 dans la salle simulée. Gris : vraie FRF. Noir : estimée avec Autostep après 60 s. . . . .	45
3.12	Dispositif expérimental installé dans la salle semi-anéchoïque. . . . .	46
3.13	Réponse impulsionnelle du haut-parleur 1 dans la salle semi-anéchoïque. Gris : mesurée avec un signal ESS. Noir : estimée avec MISO-AS après 60 s ( $\kappa = 0.01$ , $\gamma = 10$ ). Un zoom sur le début de la RI est proposé. L'erreur $\epsilon$ est également fournie en bas du graphique (axe des ordonnées à droite de la figure). . . . .	47
3.14	FRF (module) du haut-parleur 1 dans la salle semi-anéchoïque. Gris : mesurée avec un signal ESS. Noir : estimée avec MISO-AS après 60 s. . . . .	47
3.15	Setup expérimental dans la salle du laboratoire. . . . .	48
3.16	Réponse impulsionnelle du haut-parleur 1 dans la salle du laboratoire. Gris : mesurée avec un signal ESS. Noir : estimée avec MISO-AS après 60 s ( $\kappa = 0.01$ , $\gamma = 10$ ). Un zoom sur le début de la RI est proposé. L'erreur $\epsilon$ est également fournie en bas du graphique (axe des ordonnées à droite de la figure). . . . .	49
3.17	Réponse impulsionnelle du haut-parleur 8 dans la salle du laboratoire. Gris : mesurée avec un signal ESS. Noir : estimée avec MISO-AS après 60 s ( $\kappa = 0.01$ , $\gamma = 10$ ). Un zoom sur le début de la RI est proposé. L'erreur $\epsilon$ est également fournie en bas du graphique (axe des ordonnées à droite de la figure). . . . .	50
3.18	FRF (module) du haut-parleur 1 dans la salle du laboratoire. Gris : mesurée avec un signal ESS. Noir : estimée avec MISO-AS après 6 min. . . . .	50
3.19	FRF (module) du haut-parleur 8 dans la salle du laboratoire. Gris : mesurée avec un signal ESS. Noir : estimée avec MISO-AS après 6 min. . . . .	51
4.1	<i>Learning curves</i> avec $K = 100$ dans (3.20) pour un changement brusque. Gris : MISO-LMS avec $\mu^{(l)} = 1/(NL + 1)$ . Noir : MISO-Autostep avec $\kappa = 0.01$ , $\gamma = 10$ . . . . .	55

6.1	Schéma de principe pour les <i>sound zones</i> . . . . .	59
7.1	Géométrie considérée dans ce chapitre. . . . .	80
7.2	Dispositif expérimental installé dans la salle semi-anéchoïque. . . . .	81
7.3	Erreur de convergence BACC-PM-Gd et BACC-PM-Gdc. . . . .	83
7.4	Contraste en fonction de la fréquence des méthodes BACC-PM, BACC-PM-Gd et BACC-PM-Gdc après 2000 itérations. Les courbes BACC-PM et BACC-PM-Gdc sont superposées. . . . .	85
7.5	Erreur de reproduction au cours du temps des méthodes BACC-PM, BACC-PM-Gd et BACC-PM-Gdc après 2000 itérations. Les courbes BACC-PM et BACC-PM-Gdc sont superposées. . . . .	86
7.6	Filtre appliqué au haut-parleur $l = 0$ pour les méthodes BACC-PM, BACC-PM-Gd et BACC-PM-Gdc après 2000 itérations. Les courbes BACC-PM et BACC-PM-Gdc sont superposées. . . . .	87
7.7	Contraste en fonction de la fréquence des méthodes BACC-PM, BACC-PM-Gd et BACC-PM-Gdc après 100 itérations. . . . .	87
7.8	Erreur de reproduction au cours du temps des méthodes BACC-PM, BACC-PM-Gd et BACC-PM-Gdc après 100 itérations. Les courbes BACC-PM et BACC-PM-Gdc sont superposées. . . . .	88
7.9	Erreur de convergence BACC-PM-Gd et BACC-PM-Gdc obtenue à partir des réponses impulsionnelles mesurées. . . . .	88
7.10	Contraste en fonction de la fréquence des méthodes BACC-PM, BACC-PM-Gd et BACC-PM-Gdc après 2000 itérations en mesure. Les courbes BACC-PM et BACC-PM-Gdc sont superposées. . . . .	89
7.11	Erreur de reproduction au cours du temps des méthodes BACC-PM, BACC-PM-Gd et BACC-PM-Gdc après 2000 itérations obtenue à partir des réponses impulsionnelles mesurées. Les courbes BACC-PM et BACC-PM-Gdc sont superposées. . . . .	89
8.1	Géométrie de la simulation d'inversion de zones à $t = 0$ s. Le HP 2 permet de générer la pression cible au départ, puis le HP 9 génèrera la pression cible après l'inversion des zones. . . . .	92
8.2	Contraste et erreur de reproduction au cours du temps pour l'algorithme BACC-PM-Gd. Les zones sont interverties à $t = 0,5$ s. . . . .	94

TABLE DES FIGURES

---

8.3	Contraste en fonction de la fréquence à différents instants pour l'algorithme BACC-PM-Gd. . . . .	95
8.4	Filtre du haut-parleur 2 de la géométrie 8.1 à différents instants pour l'algorithme BACC-PM-Gd. . . . .	96
8.5	Contraste et erreur de reproduction au cours du temps pour l'algorithme BACC-PM-Gdc. Les zones sont interverties à $t = 0,5$ s. . . . .	97
8.6	Contraste en fonction de la fréquence à différents instants pour l'algorithme BACC-PM-Gdc. . . . .	98
8.7	Filtre du haut-parleur 2 de la géométrie 8.1 à différents instants pour l'algorithme BACC-PM-Gdc. . . . .	99
8.8	Géométrie de la simulation lorsque la <i>bright zone</i> avance vers la droite. Le HP 4 permet de générer la pression cible. . . . .	100
8.9	Contraste moyen et erreur moyenne au cours du temps lorsque la <i>bright zone</i> avance à $1 \text{ m s}^{-1}$ vers la droite. A $t = 0$ s les filtres sont initialisés par des diracs. . . . .	102
8.10	Contraste moyen au cours du temps lorsque la <i>bright zone</i> avance à $1 \text{ m s}^{-1}$ vers la droite. À $t = 0$ s les filtres sont initialisés par les filtres générés par inversion directe. . . . .	103
8.11	Contraste moyen au cours du temps lorsque la <i>bright zone</i> avance à $0,5 \text{ m s}^{-1}$ vers la droite. À $t = 0$ s les filtres sont initialisés par les filtres générés par inversion directe. . . . .	104
8.12	Contraste moyen et erreur moyenne au cours du temps lorsque la <i>bright zone</i> avance à $1 \text{ m s}^{-1}$ vers la droite. A $t = 0$ les filtres sont initialisés par des diracs. Les RI sont mises à jour tous les 10 cm. . . . .	105
9.1	Contraste et erreur de reproduction au cours du temps pour l'algorithme BACC-PM-Gd et $M$ MISO-Autostep dans un cas statique. . . . .	109
9.2	Contraste et erreur de reproduction au cours du temps pour l'algorithme BACC-PM-Gd et $M$ MISO-Autostep avec une inversion des zones à $t = 0,2$ s. . . . .	110
9.3	Contraste et erreur de reproduction au cours du temps pour l'algorithme BACC-PM-FxLMS. . . . .	112
9.4	Contraste en fonction de la fréquence pour l'algorithme BACC-PM-FxLMS à la fin de la simulation. . . . .	113
9.5	Erreur quadratique moyenne des $M$ MISO-Autostep. . . . .	114

9.6	Contraste et erreur de reproduction pour les algorithmes MISO-Autostep et BACC-PM-FxLMS. . . . .	115
9.7	Contraste et erreur de reproduction pour le Fx-LMS avec MISO-Autostep dans le cas de conditions initiales mesurées. . . . .	116
B.1	Géométrie optimisée . . . . .	127



# LISTE DES TABLEAUX

---

3.1	Médiane du temps de calcul (sur 100 essais)   Intel i7-10875H @ 2,3 GHz   64 Gb RAM   Script Matlab . . . . .	41
3.2	Désalignement et erreur spectrale quadratique moyenne dans la salle simulée après 60 s. Moyenne sur tous les haut-parleurs, valeurs min et max. . . . .	45
3.3	Désalignement et erreur spectrale quadratique moyenne dans la salle semi-anéchoïque après 6 s. Moyenne sur tous les haut-parleurs, valeurs min et max. . . . .	48
3.4	Désalignement et erreur spectrale quadratique moyenne dans la salle du laboratoire après 6 min. Moyenne sur tous les haut-parleurs, valeurs min et max. . . . .	49
7.1	Effort du réseau des méthodes BACC-PM, BACC- PM-Gd et BACC-PM-Gdc après 2000 itérations. . . . .	84
7.2	Effort du réseau des méthodes BACC-PM, BACC- PM-Gd et BACC-PM-Gdc après 100 itérations. . . . .	85
7.3	Effort du réseau des méthodes BACC-PM, BACC- PM-Gd et BACC-PM-Gdc après 2000 itérations obtenu à partir des réponses impulsionnelles mesurées. . . . .	86



# NOTATIONS

---

- $(\cdot)^+$  Fonction partie positive
- $(\cdot)^\top$  Transposée
- $(\cdot)^{(l)}$  Valeur liée au haut-parleur  $l$
- $(\cdot)^{(m)}$  Valeur liée au microphone  $m$
- $(\cdot)^{(m,l)}$  Valeur liée au microphone  $m$  et au haut-parleur  $l$
- $(\cdot)_B$  Valeur liée à la *bright zone*
- $(\cdot)_D$  Valeur liée à la *dark zone*
- $(\circ)$  Multiplication élément par élément
- $(\oslash)$  Division élément par élément
- A** Matrice
- a** Vecteur
- $\mathbf{A}_{M \times N}$**  Matrice de  $M$  lignes et  $N$  colonnes
- $\lfloor \cdot \rfloor$  Arrondi à l'entier inférieur le plus proche
- $\text{Tr}[\cdot]$  Trace de la matrice
- $\| \cdot \|$  Norme  $\ell_2$
- $|\cdot|$  Norme  $\ell_1$ , module pour les valeurs complexes
- $a$  Scalaire
- $f[\mathbf{a}]$  Vecteur  $[f(a_i)]$  dont la fonction  $f$  est appliquée à chaque élément du vecteur  $\mathbf{a}$





# INTRODUCTION

---

## Contexte et introduction aux *sound zones*

L'enregistrement et la reproduction d'une onde acoustique est un sujet de recherche ancien qui ne cesse d'évoluer. En 1993, BERKHOUT *et al.* [1] puis KIBERBEY *et al.* [2] proposent une théorie, basée sur la théorie de KIRCHHOFF-HELMHOLTZ, de reproduction d'un front d'onde continu à partir d'un réseau de haut-parleurs et de quelques points de mesure. Cette théorie, appelée *Wave Field Synthesis* [3], permet en particulier de résoudre des problématiques de nuisance sonore. En effet, avec le développement du filtrage numérique, il est possible de construire des systèmes d'annulation de bruit [4], où quelques microphones et quelques haut-parleurs suffisent à diminuer le niveau sonore dans une partie de l'espace donné. Un système d'annulation de bruit étant équivalent à un système de création de champ sonore, KIRBEY *et al.* [5] proposent en 1996 un principe de reconstruction d'un champ sonore dans un espace restreint à partir d'un réseau de haut-parleurs. La démarche adoptée et la théorie développée relèvent du domaine temporel.

En 1997, DRUYVESTEYN *et al.* [6] proposent trois solutions, pour trois bandes de fréquences différentes, de contrôle de la pression acoustique dans une salle. Les basses fréquences, de part leur longueur d'onde grande, sont traitées par contrôle actif. Les travaux présentés dans ce manuscrit s'inscrivent dans la continuité de ces travaux et ne s'appliquent qu'aux basses fréquences également.

En 2002, CHOI *et al.* [7] nomment *bright zone* une zone de l'espace dans laquelle le niveau acoustique est plus important que dans le reste de l'espace, *dark zone* une zone située autour de la *bright zone* dont le niveau acoustique est le plus faible possible et *zone of interest* la somme des deux zones. Une idée développée est de trouver les commandes à appliquer au réseau de haut-parleurs pour maximiser le rapport entre l'énergie potentielle acoustique au sein de la *bright zone* et le reste de l'espace (rapport nommé contraste). Tout ceci s'effectue sous contrainte, puisque, par exemple, les haut-parleurs ont une puissance d'entrée limitée.

En même temps que cette théorie se développe, la reproduction de contenus audio spatialisés s’amplifie dans l’audiovisuel grand public (cinéma, télévision, réalité virtuelle, concert, etc.), la domotique et les industries (transport, etc.). Lorsqu’ils entrent en concurrence dans un même lieu, ils nécessitent alors l’utilisation de systèmes de reproduction spécifiques. Ils ne peuvent pas être reproduits directement pour chaque auditeur sans l’utilisation d’un casque, isolant les auditeurs de leur environnement.

Les *sound zones* [8] ont pour vocation de répondre à cette problématique en minimisant l’énergie potentielle acoustique autour d’un sujet 1 (*dark zone*) tout en reproduisant le signal audio pour le sujet 2 (*bright zone*) qui souhaite écouter un programme donné. Si le sujet 1 souhaite par ailleurs écouter un autre programme, par superposition, le problème à résoudre reste le même, mais en inversant le rôle des zones. Cette capacité à reproduire des contenus audio indépendamment les uns des autres, en différents points de l’espace et suivant des critères donnés, permet de se passer de casques et ainsi de conserver une certaine convivialité tout en maintenant la conversation possible. Les applications de ces méthodes de *sound zones* existent dans différents types d’environnements comme les véhicules terrestres [9], les avions [10] voire pour d’autres applications comme la téléphonie mobile [11] ou les concerts [12].

Toutes les méthodes proposées supposent connus les chemins acoustiques entre sources et microphones (modélisés ou mesurés). En particulier, les méthodes de restitution du champ sonore utilisées dans une salle doivent prendre en compte les effets dus aux réflexions sur les parois, afin de reproduire le champ sonore souhaité avec les meilleures performances possibles.

Dans ce présent document, nous partons des travaux —dans le domaine temporel— de KIRBEY *et al.* [5] pour les appliquer aux *sound zones* et nous proposons de les étendre à des problématiques dynamiques. Enfin, nous prenons en compte les effets de salle pour rendre notre modèle utilisable dans le plus grand nombre de contextes possible. Une contrainte majeure dans tout le travail est de pouvoir exploiter les algorithmes en temps réel.

## Organisation du document et notations

Le manuscrit est composé de deux grandes parties : l'estimation de réponses impulsionnelles dans la partie I et les *sound zones* mobiles dans la partie II. Les deux parties sont lisibles de manière indépendante car les notions importantes sont rappelées en début de chapitre.

C'est le système d'unités mètre-kilogramme-seconde-ampère (MKSA) qui est retenu ici. Par ailleurs, dans tout le document, le temps est discret et l'indice  $n$  correspond à l'instant courant du temps discret.

## Liens internet

Un dépôt numérique contenant les codes utilisés dans ce document, ainsi que les campagnes de mesures est disponible sur Zenodo sous le DOI [10.5281/zenodo.11403238](https://doi.org/10.5281/zenodo.11403238).

Les fonctions proposées sont disponibles sous licence Creative Common CC BY-SA 4.0.

## Glossaire

$c_0$  : vitesse du son dans l'air

$\delta(\cdot)$  : impulsion unitaire de Dirac

$\mathbf{I}$  : matrice identité

$i$  : nombre imaginaire pur

$\omega$  : pulsation

$p$  : pression

$\rho$  : densité de l'air

$L$  : nombre de haut-parleurs

$M_B$  : nombre de microphones dans la *bright zone*

$M_D$  : nombre de microphones dans la *dark zone*

$N$  : nombre de coefficients dans les filtres de *sound zones*

$M$  : nombre total de microphones,  $M = M_B + M_D$

$K$  : nombre d'échantillons dans le signal

$J$  : fonction de coût

$J_{\min}$  : minimum de la fonction coût

$p^{(m)}$  : pression mesurée par le microphone  $m$

$p_t$  : pression cible (*target pressure*)

$d$  : signal désiré (*desired signal*)

$y^{(l)}$  : signal alimentant le haut-parleur  $l$

$h_r^{(m,l)}$  : réponse impulsionnelle réelle entre le microphone  $m$  et le haut-parleur  $l$

$h^{(m,l)}$  : réponse impulsionnelle estimée entre le microphone  $m$  et le haut-parleur  $l$

$w^{(l)}$  : filtre de *sound zones* pour le haut-parleur  $l$

$e$  : signal d'erreur

$n$  : numéro de l'échantillon à temps discret

$k$  : numéro de l'itération à temps discret

$\mu, \mu_k$  : pas de descente du gradient, pas de descente du gradient variant dans le temps  
(1<sup>re</sup> partie)

$\theta_k$  : pas de descente du gradient (2<sup>e</sup> partie)

$\tilde{\mu}$  : pas de descente normalisé du gradient

$x$  : signal d'entrée

$T_s$  : période d'échantillonnage

$F_s$  : fréquence d'échantillonnage  $F_s = 1/T_s$

$\kappa, \gamma$  : méta-paramètres

$\lambda$  : paramètre de régularisation

$\tau$  : constante de temps

$\epsilon$  : petit paramètre

$e_P, e_B, e_D$  : énergie potentielle, énergie potentielle dans la *bright zone*, énergie potentielle dans la *dark zone*

---

## Abréviations

**SISO** : *Single Input Single Output* (une entrée une sortie)

**MISO** : *Multiple Input Single Output* (plusieurs entrées plusieurs sorties)

**MIMO** : *Multiple Input Multiple Output* (plusieurs entrées une sortie)

**LTI** : *Linear Time-Invariant* (linéaire invariant dans le temps)

**LTV** : *Linear Time-Variant* (linéaire variant dans le temps)

**LMS** : *Least Mean Square* (moindres carrés instantanés)

**NLMS** : *Normalised Least Mean Square* (moindres carrés instantanés normalisés)

**RLS** : *Recursive Least Square* (moindres carrés récursifs)

**FIR** : *Finite Impulse Response* (réponse impulsionnelle finie)

**iid** : *independant identical distributed* (indépendant identiquement distribué)

**mis** : *Misalignment* (désajustement)

**mse** : *Mean square error* (erreur quadratique moyenne)

**mse<sub>DFT</sub>** : *Spectral mean square error* (erreur quadratique moyenne fréquentielle)

**RT** : *Reverberation time* (temps de réverbération)



PREMIÈRE PARTIE

# Estimation de réponses impulsionnelles

---



# POSITION DU PROBLÈME

Un système audio dédié à la création de *sound zones* utilise de nombreuses sources électroacoustiques (haut-parleurs) situées autour de la zone de l'espace où le champ acoustique doit être contrôlé (figure 1.1). Dans cet espace, deux zones sont habituellement définies, une zone d'écoute nommée *bright zone* et une zone silencieuse nommée *dark zone*, les algorithmes de *sound zones* ayant alors pour objet de filtrer les signaux émis par les sources en fonction des contraintes liées à ces deux zones. Par ailleurs, pour diffuser deux contenus audio différents dans les deux zones, il suffit d'inverser les zones préalablement définies et de résoudre à nouveau le problème : par superposition, le résultat à deux contenus audio est la somme de deux sous-problèmes, chacun se résumant à une *bright zone* et une *dark zone*. Chaque zone est échantillonnée par une antenne de microphones. Ces antennes permettent de générer les filtres de *sound zones* et de vérifier la création des zones. L'ensemble du dispositif peut donc se modéliser comme un système *Multiple Input Multiple Output* (MIMO).

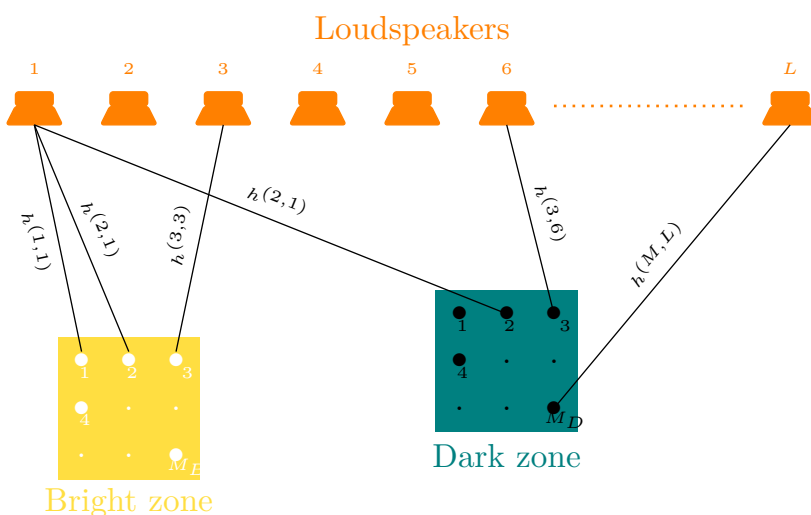


FIGURE 1.1 – Schéma de principe d'un système audio de *sound zones*.

## 1.1 Sound zones et réponses impulsionnelles

Tous les algorithmes de *sound zones* (voir partie II) nécessitent une connaissance *a priori* des réponses impulsionnelles (ou des réponses en fréquence) de toutes les combinaisons « haut-parleur / microphone » du système MIMO. Nous noterons par la suite la réponse impulsionnelle entre le microphone  $m$  et le haut-parleur  $l$  :  $h^{(m,l)}$ . À partir de ces informations, il est possible d'estimer les filtres à appliquer à chaque haut-parleur pour créer un champ sonore répondant aux contraintes voulues par l'utilisateur.

L'équation qui relie la pression mesurée au microphone  $m \in [1, M]$  et notée  $p^{(m)}(n)$ , aux signaux d'entrée  $y^{(l)}(n)$  des  $L$  haut-parleurs est <sup>1</sup> :

$$p^{(m)}(n) = \sum_{l=1}^L \left( h^{(m,l)} * y^{(l)} \right) (n), \quad (1.1)$$

cette pression mesurée étant la somme des contributions de chaque haut-parleur. Les quantités  $m$ ,  $l$  et  $n$  désignent respectivement les indices du microphone, du haut-parleur et du temps discret. De plus, le signal d'entrée du haut-parleur  $l$  est une version du signal à diffuser  $x(n)$  convoluée par un filtre de *sound zones* noté  $w^{(l)}(n)$  :

$$y^{(l)}(n) = \left( w^{(l)} * x \right) (n). \quad (1.2)$$

Ainsi, l'équation générale du système s'écrit sous la forme :

$$p^{(m)}(n) = \sum_{l=1}^L \left( h^{(m,l)} * w^{(l)} * x \right) (n). \quad (1.3)$$

Dans cette partie, nous considérons connus les signaux émis par les haut-parleurs et mesurés par les microphones, et nous cherchons à déterminer les réponses impulsionnelles  $h^{(m,l)}(n)$ .

## 1.2 Réduction du système MIMO à M systèmes MISO

Comme évoqué dans la section précédente, le système à estimer est MIMO puisqu'il possède plusieurs entrées (les haut-parleurs) et plusieurs sorties (les microphones). Cependant, l'équation 1.3 indique que chaque mesure microphonique est indépendante des

---

1. Le bruit de mesure est considéré comme négligeable par rapport aux différents signaux du système.

autres. Il est donc possible de traiter le problème comme une somme de  $M$  systèmes MISO (*Multiple Input Single Output*) [13].

Le signal émis par chaque haut-parleur est une version du signal à diffuser,  $x(n)$ , filtrée par les différents filtres de *sound zones*,  $w^{(l)}(n)$ . Ceci est valable lors de l'utilisation du système, en exploitation. Cependant, une phase de calibration permet de piloter individuellement chaque haut-parleur avec des signaux  $y^{(l)}(n)$  conçus spécifiquement pour cette phase. Si chaque haut-parleur peut être piloté indépendamment des autres, il est donc possible de subdiviser chaque problème MISO en une somme de problèmes SISO (*Single Input Single Output*) indépendants. Cette décomposition n'est cependant pas possible en exploitation, le signal émis étant commun à tous les haut-parleurs.

# ÉTAT DE L'ART

---

L'estimation de réponses impulsionnelles de systèmes multicanaux est au cœur de nombreuses applications audio comme en synthèse de champs, *Wave Field Synthesis* (WFS) [1], en ambisonie d'ordres supérieurs (*High Order Ambisonics* - HOA) [14] ou pour l'implémentation de *sound zones* [7], [14]-[16]. Les systèmes de compensation d'effet de salle [17], [18], de contrôle actif de bruit [19] ou d'annulation d'écho [20] peuvent également être considérés comme des applications classiques nécessitant une estimation de réponses impulsionnelles de systèmes multicanaux. Dans le cadre de ces applications, l'estimation *in situ* des réponses impulsionnelles est un facteur clef pour optimiser les performances des algorithmes et/ou pour contrôler le champ acoustique.

Si le système est SISO et invariant dans le temps, les réponses impulsionnelles peuvent être mesurées par diverses méthodes *off-line*, par exemple en utilisant des signaux MLS (*Maximum Length Sequence*) [21] ou des signaux ESS (*Exponential Swept Sine*) [22] pour les plus connus (cf. section 2.1). Lorsque le système possède plusieurs entrées (MISO ou MIMO) et est invariant dans le temps, des techniques à base de signaux orthogonaux (cf. section 2.2) peuvent être utilisées [23], [24].

Toutes ces méthodes sont utilisables dans un contexte de *post-processing*, où les signaux sont exploités une fois la mesure complète réalisée. Au contraire, les algorithmes adaptatifs permettent, quant à eux, de traiter les informations dès qu'elles sont disponibles et peuvent estimer en temps réel les réponses impulsionnelles.

Par ailleurs, l'incorporation de nouvelles données à un instant donné rend généralement les algorithmes adaptatifs peu gourmands en termes de mémoire, puisque seuls les derniers points de mesures sont utiles aux calculs. De plus, le caractère récursif des algorithmes leur permet de s'adapter au cours du temps et donc de pouvoir estimer au besoin des réponses impulsionnelles de systèmes variant dans le temps.

Une part importante des algorithmes adaptatifs de la littérature est basée sur l'algorithme LMS (*Least Mean Square*), où les coefficients du filtre sont calculés par minimisation de l'erreur quadratique instantanée [25]. Le processus d'optimisation est lié à une descente de gradient stochastique, le filtre étant adapté en prenant seulement en compte l'erreur à l'instant courant. Pour moduler la descente vers le minimum global, le filtre LMS utilise un pas de descente scalaire,  $\mu > 0$ , qui dépend du système étudié (le paramètre étant homogène à une énergie). La version normalisée de l'algorithme LMS, notée NLMS, consiste à normaliser le paramètre  $\mu$  par l'énergie instantanée du signal d'entrée.

Enfin, d'autres méthodes dites "méta-gradient" proposent de calculer un vecteur  $\mu$  par descente de gradient. SUTTON propose en ce sens l'algorithme IDBD (*Incremental Delta-Bar-Delta*) [26] : l'approche consiste à utiliser différents pas de descente de gradient, en fonction des éléments qui composent l'estimation de la réponse impulsionnelle et l'erreur à l'instant courant. Par la suite, MAHMOOD [27] propose de normaliser ce vecteur  $\mu$  pour qu'il ne dépende plus du système physique mesuré. C'est cette méthode, appelée Autostep, qui a été étendue aux systèmes MISO et appliquée à l'estimation de réponses impulsionnelles de salles dans le cadre de la première partie de ce document. Les méthodes adaptatives utilisées en contexte SISO sont décrites en section 2.3.

## 2.1 *Exponential swept sine* pour système SISO

Dans le contexte des systèmes SISO invariant dans le temps, plusieurs méthodes d'estimation *off line* de réponse impulsionnelle sont disponibles dans la littérature [28], [29], basées sur des choix de signaux d'excitation particuliers. Parmi ces méthodes, celles basées sur des excitations du type *chirp* ou *swept sine* sont considérées parmi les plus intéressantes, en particulier en termes de contrôle de paramètres d'entrée (choix de la loi de variation de la fréquence instantanée) et de possibilité d'écriture analytique du filtre inverse dans le domaine temporel [28], [29]. Plus précisément, la méthode ESS (pour *exponential swept sine*) sera considérée par la suite comme la méthode de référence dans le champ non adaptatif *off line*. Cette technique de mesure a été proposée par FARINA [22] puis modifiée par NOVAK *et al.* [30].

Si un seul haut-parleur est actif à la fois et s'il n'y a pas de filtrage pour les *sound zones*, l'équation (1.3) devient :

$$p^{(m)}(n) = \left( h^{(m,l)} * y^{(l)} \right) (n), \quad (2.1)$$

avec les signaux émis et reçu connus (i.e. le signal mesuré  $p^{(m)}$  et le signal généré  $y^{(l)}$ ), et la réponse impulsionnelle  $h^{(m,l)}$  inconnue. Le signal d'excitation est une sinusoïde dont la fréquence instantanée varie de manière exponentielle avec le temps :

$$y^{(l)}(n) = \sin \left( A e^{nT_s/B} \right), \quad (2.2)$$

où

$$A = 2\pi f_1 B \text{ et } B = \frac{T}{\log f_2 - \log f_1},$$

avec  $T_s$  la période d'échantillonnage,  $T$  la durée du signal,  $f_1$  et  $f_2$  les fréquences respectivement de départ et de fin du signal.

L'expression analytique de l'inverse de la transformée de Fourier du signal à temps continu  $y^{(l)}(t)$  étant connu [30], il est possible de générer un signal  $\tilde{y}_{ess}^{(l)}(n)$  tel que :

$$\left( y^{(l)} * \tilde{y}_{ess}^{(l)} \right) (n) = \delta(n), \quad (2.3)$$

avec

$$\tilde{y}_{ess}^{(l)}(n) = \frac{f_1}{B} e^{-nT_s/B} y^{(l)}(-n). \quad (2.4)$$

Sous condition d'un rapport signal-sur-bruit élevé, la réponse impulsionnelle est alors calculée par convolution linéaire entre le signal mesuré et le signal  $\tilde{y}_{ess}^{(l)}(n)$  suivant :

$$h^{(m,l)}(n) = \left( p^{(m)} * \tilde{y}_{ess}^{(l)} \right) (n) = \left( h^{(m,l)} * y^{(l)} * \tilde{y}_{ess}^{(l)} \right) (n). \quad (2.5)$$

Les supports temporels des signaux émis et inverse doivent être cohérents avec le support temporel de la réponse impulsionnelle à mesurer, des zéros pouvant être ajoutés à la fin des signaux pour obtenir la fin de la réponse impulsionnelle.

## 2.2 Signaux orthogonaux pour systèmes MISO

Dans le cas d'un système MISO invariant dans le temps, il est possible d'actionner simultanément les  $L$  sources afin d'estimer en même temps les  $L$  réponses impulsionnelles

[31].

Parmi les techniques de mesure de systèmes MISO, celles à base de signaux orthogonaux utilisent la propriété de décorrélation entre deux signaux. En effet, si deux signaux sont émis par deux haut-parleurs différents, il est possible de déduire quel haut-parleur émet quel signal et ainsi d'en déduire la réponse impulsionnelle entre chaque haut-parleur indépendamment des autres. Les signaux peuvent être orthogonaux dans le domaine temporel (cas de deux bruits blancs dont l'intercorrélacion est nulle) ou dans le domaine fréquentiel (cas de signaux multi-tons dont l'interspectre est nul).

Notons qu'il existe également des méthodes qui accélèrent les temps de mesure à partir de signaux composites non strictement orthogonaux, comme les *sweeps* entrelacés [32].

L'avantage de ces méthodes est souvent le temps de mesure qui est plus court que celui nécessaire à la mesure indépendante de toutes les réponses impulsionnelles individuelles, même s'il reste toutefois incompatible avec des applications qui visent le temps réel [33].

## 2.3 Méthodes adaptatives en contexte SISO

Les méthodes adaptatives se basent sur un critère d'erreur et une méthode d'optimisation qui a pour objectif d'estimer les coefficients du filtre qui minimise le critère d'erreur. Cette estimation est réalisée au cours du temps, à mesure que les signaux sont émis et reçus. Cette capacité à travailler au cours du temps, échantillon par échantillon, ouvre par ailleurs la voie à la caractérisation de systèmes variant dans le temps, comme c'est le cas lorsque les positions des *sound zones* bougent. L'ouvrage de référence abondamment utilisé et cité dans ce document est celui écrit par Simon HAYKIN, *Adaptive Filter Theory* [25].

Les méthodes les plus populaires parmi les méthodes adaptatives sont :

- l'algorithme RLS,
- les méthodes dérivées de l'algorithme LMS,
- les filtres de Kalman.

De manière générale, le filtre RLS présente un temps de convergence plus faible que le filtre LMS, pour un coût de calcul plus important. Or, les réponses impulsionnelles de salle peuvent être longues et le coût de calcul d'un filtre RLS étant en  $O(N^2)$  [34], le temps de calcul résultant peut s'avérer prohibitif. De plus, en régime permanent, le filtre RLS n'est pas plus performant qu'un filtre LMS [25]. Des algorithmes Newton-Gauss déterminent le pas optimal  $\mu$  en utilisant l'inverse de la matrice Hessienne [35]. Cependant, ces techniques souffrent d'un coût de calcul élevé, principalement dû à l'inversion de la matrice Hessienne [34]. Des versions dites Quasi-Newton de ces algorithmes tentent d'estimer l'inverse de l'Hessienne comme l'algorithme *quasi-Newton stochastic gradient descent* (SGQ-QN) [36]. Par ailleurs, le filtrage de Kalman nécessite une modélisation du problème sous forme d'équation d'état. À ce jour et au regard de la grande variété des formes de réponses impulsionnelles de salle, il n'existe pas d'équation d'état décrivant l'évolution instantanée de la réponse impulsionnelle d'une salle au cours du temps. Par conséquent, le filtrage de Kalman a également été mis de côté pour l'étude présentée ici et c'est donc autour d'une architecture de type LMS qu'a été construite la méthode développée dans cette partie pour estimer les réponses impulsionnelles de systèmes MISO.

Par souci de simplification, les notations dans cette partie seront les suivantes : réponse impulsionnelle à estimer  $h_r(n)$ , signal d'excitation du haut-parleur  $x(n)$ <sup>1</sup> et signal mesuré par le microphone  $p(n)$ . Le contexte est celui d'un système SISO, l'extension aux problèmes MISO étant traitée dans la section suivante 2.4.

Sur la figure 2.1, la réponse impulsionnelle à estimer est donc notée  $h_r(n)$  et le signal d'excitation choisi (et donc connu) par l'utilisateur est noté  $x(n)$ . Le signal mesuré, bruité par  $b(n)$ , est noté  $p(n)$  et est également connu. Le signal « désiré »  $d(n)$  est inconnu et inaccessible. Un filtre  $h(n)$  est placé en parallèle du système à identifier dont l'algorithme adaptatif pourra modifier les coefficients. En sortie de ce filtre, le signal est noté  $\hat{p}(n)$  (la notation  $\hat{p}$  signifie l'estimation de  $p$ ). En l'absence de bruit, si  $h(n) = h_r(n)$ , alors le signal mesuré est égal au signal estimé et l'erreur est donc parfaitement nulle.

L'enjeu ici est donc de trouver le filtre  $h(n)$  tel que l'erreur soit minimale.

---

1. Nous considérons ici que les filtres de *sound zones* sont inactifs, donc  $y(n) = (\delta * x)(n) = x(n)$ . Le choix d'utiliser  $x$  pour le signal d'entrée et non  $y$  permet de rester cohérent avec la littérature existante.



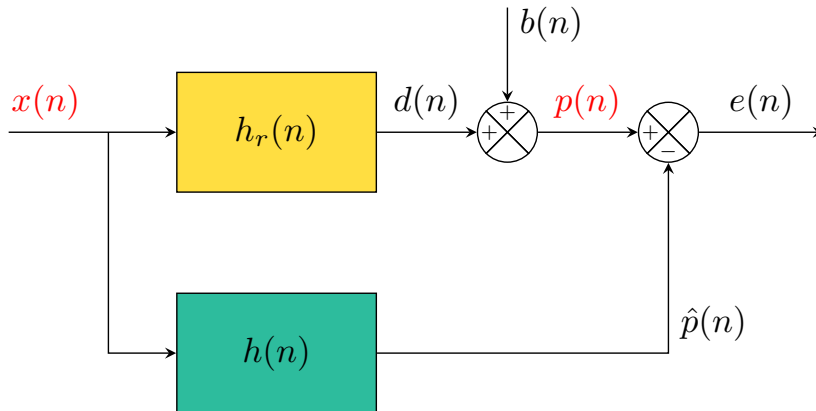


FIGURE 2.1 – Schéma bloc d'un filtre adaptatif pour l'estimation de réponses impulsionnelles. En rouge les signaux connus, en jaune la réponse impulsionnelle à estimer et en vert le filtre que l'algorithme adaptatif peut modifier.

La réponse impulsionnelle est considérée par la suite à support temporel fini, ce qui permet de choisir le filtre  $h(n)$  comme un filtre FIR de  $N$  coefficients, qu'il est habituel de regrouper sous forme de vecteur :

$$\mathbf{h} = [h_0 \ \cdots \ h_{N-1}]^\top, \quad (2.6)$$

avec  $(\cdot)^\top$  la transposée.

De même, pour le signal d'entrée, nous utiliserons indifféremment la notation  $\mathbf{x}(n)$  ou  $\mathbf{x}_n$ , qui correspond aux  $N$  derniers échantillons du signal d'entrée :

$$\mathbf{x}_n = [x(n) \ x(n-1) \ \cdots \ x(n-N+1)]^\top. \quad (2.7)$$

Ces notations permettent d'écrire les convolutions sous forme vectorielle puisque l'échantillon à l'instant  $n$  qui sort du filtre  $h(n)$ , lorsqu'il est excité par le signal  $\mathbf{x}_n$ , se note :

$$p(n) = \mathbf{h}^\top \mathbf{x}_n. \quad (2.8)$$

Par la suite, le choix de l'ordre  $N$  est un compromis qui doit permettre d'englober toute la réponse impulsionnelle du système, sans être trop élevé car le coût de calcul augmente avec l'ordre.

### 2.3.1 Filtrage de Wiener en contexte stationnaire

Dans le cas où le système est linéaire et invariant dans le temps, la fonction de coût  $J$ , forme quadratique de l'erreur d'estimation, s'écrit dans un contexte stochastique pour des signaux stationnaires [37] sous la forme :

$$J(\mathbf{h}) = \mathbb{E}[e^2(n)] = \mathbb{E}\left[\left(p(n) - \mathbf{h}^\top \mathbf{x}_n\right)^2\right], \quad (2.9)$$

la notation  $\mathbb{E}[\cdot]$  représentant l'espérance mathématique. Le développement de cette expression permet d'obtenir :

$$J(\mathbf{h}) = \mathbb{E}\left[p^2(n)\right] - 2\mathbf{h}^\top \mathbb{E}\left[\mathbf{x}_n p(n)\right] + \mathbf{h} \mathbb{E}\left[\mathbf{x}_n \mathbf{x}_n^\top\right] \mathbf{h}^\top. \quad (2.10)$$

En notant

$$\mathbf{R} = \mathbb{E}\left[\mathbf{x}_n \mathbf{x}_n^\top\right]$$

la matrice de corrélation du signal d'entrée (de taille  $N \times N$ ), symétrique et semi-définie positive,

$$\boldsymbol{\rho} = \mathbb{E}\left[\mathbf{x}_n p(n)\right]$$

le vecteur d'intercorrélation entre l'entrée et la sortie (de taille  $N \times 1$ ) et

$$\sigma_p^2 = \mathbb{E}\left[p^2(n)\right]$$

la variance<sup>2</sup> du signal de sortie, la fonction de coût s'écrit alors :

$$J(\mathbf{h}) = \sigma_p^2 - 2\mathbf{h}^\top \boldsymbol{\rho} + \mathbf{h}^\top \mathbf{R} \mathbf{h}. \quad (2.11)$$

La fonction de coût étant quadratique, elle est donc minimale lorsque le gradient s'annule :

$$\nabla J = -2\boldsymbol{\rho} + 2\mathbf{R} \mathbf{h} = 0, \quad (2.12)$$

ce qui conduit alors à l'équation Wiener-Hopf sous forme matricielle<sup>3</sup> :

$$\mathbf{h}_o = \mathbf{R}^{-1} \boldsymbol{\rho}. \quad (2.13)$$

---

2. Les signaux acoustiques étant tous centrés.

3. La matrice  $\mathbf{R}$  étant inversible, car semi-définie positive.

Nous trouvons alors une valeur minimale pour la fonction de coût :

$$J_{\min} = \sigma_p^2 - \boldsymbol{\varrho}^\top \mathbf{R}^{-1} \boldsymbol{\varrho}. \quad (2.14)$$

Le filtre optimal  $h_o(n)$  est donc le filtre qui minimise l'erreur quadratique moyenne, ou, de manière équivalente, celui qui estime au mieux - au sens des moindres carrés - la réponse impulsionnelle  $h_r(n)$ . Cependant, d'un point de vue pratique, résoudre l'équation 2.13 impose de connaître  $\mathbf{R}$  la fonction d'autocorrélation du signal d'entrée et  $\boldsymbol{\varrho}$  l'intercorrélacion entre les signaux d'entrée et de sortie. De plus, comme l'autocorrélation (ou l'intercorrélacion) est calculée sur tout le signal, et ne dépend pas du temps, ce filtre n'est utilisable que dans un contexte de signaux stationnaires et de systèmes invariant dans le temps.

### 2.3.2 Algorithme *Recursive Least Square* (RLS)

Un algorithme récursif pour la conception de filtres adaptatifs à réponse impulsionnelle finie (FIR) est disponible dans la littérature : l'algorithme RLS [25]. Son principe est basé sur l'estimation récursive des différents ingrédients nécessaires à l'obtention de la solution de Wiener.

Supposons que nous ne disposons des informations que jusqu'à l'instant courant  $n$ , que nous notons :

$$\mathbf{R}_n = \frac{1}{n} \sum_{k=1}^n \mathbf{x}_k \mathbf{x}_k^\top, \quad (2.15)$$

pour l'approximation de la matrice de corrélation du signal d'entrée, et

$$\boldsymbol{\varrho}_n = \frac{1}{n} \sum_{k=1}^n \mathbf{x}_k p_k, \quad (2.16)$$

pour l'approximation du vecteur d'intercorrélacion entre entrée et sortie, avec

$$\mathbf{x}_k = \left[ x(k) \quad \cdots \quad x(k - N + 1) \right]^\top$$

le vecteur contenant les  $N$  derniers échantillons du signal d'entrée à partir de l'instant <sup>4</sup>  $k$ .

---

4. Pour les instants au début ( $k < N$ ), le vecteur  $\mathbf{x}_k$  fait appel à des valeurs de temps négatives pour le signal d'entrée. Nous supposons par causalité  $\forall k < 0, x(k) = 0$ .

Comme à chaque instant  $n$ , la matrice  $\mathbf{R}_n$  et le vecteur  $\boldsymbol{\varrho}_n$  doivent respecter les équations normales,  $\mathbf{R}_n \mathbf{h}_o = \boldsymbol{\varrho}_n$ , il est possible d'omettre le terme  $1/n$  dans les estimations de l'autocorrélation et de l'intercorrélation.

À l'instant  $n + 1$ , nous pouvons donc écrire :

$$\mathbf{R}_{n+1} = \sum_{k=1}^n \mathbf{x}_k \mathbf{x}_k^\top + \mathbf{x}_{n+1} \mathbf{x}_{n+1}^\top = \mathbf{R}_n + \mathbf{x}_{n+1} \mathbf{x}_{n+1}^\top \quad (2.17)$$

et

$$\boldsymbol{\varrho}_{n+1} = \boldsymbol{\varrho}_n + \mathbf{x}_{n+1} p_{n+1}. \quad (2.18)$$

L'équation de Wiener (2.13) s'écrit alors, à l'instant  $n + 1$ , sous la forme :

$$\mathbf{h}_{n+1} = \mathbf{R}_{n+1}^{-1} \boldsymbol{\varrho}_{n+1}. \quad (2.19)$$

Par application du lemme d'inversion matricielle<sup>5</sup> et en posant  $\Phi_n = (\mathbf{R}_n)^{-1}$ , nous obtenons :

$$\Phi_{n+1} = \Phi_n - \mathbf{k}_{n+1} \mathbf{x}_{n+1}^\top \Phi_n, \quad (2.20)$$

avec  $\mathbf{k}_{n+1}$  le gain de Kalman à l'instant  $n + 1$ , tel que :

$$\mathbf{k}_{n+1} = \frac{\Phi_n \mathbf{x}_{n+1}}{1 + \mathbf{x}_{n+1}^\top \Phi_n \mathbf{x}_{n+1}} = \Phi_{n+1} \mathbf{x}_{n+1}. \quad (2.21)$$

À l'instant  $n + 1$ , l'équation de Wiener devient donc :

$$\mathbf{h}_{n+1} = \Phi_{n+1} \boldsymbol{\varrho}_{n+1} = \Phi_{n+1} \boldsymbol{\varrho}_n + \Phi_{n+1} \mathbf{x}_{n+1} p_{n+1}, \quad (2.22)$$

soit en remplaçant  $\Phi_{n+1}$  par son expression (2.20) :

$$\mathbf{h}_{n+1} = \Phi_n \boldsymbol{\varrho}_n - \mathbf{k}_{n+1} \mathbf{x}_{n+1}^\top \Phi_n \boldsymbol{\varrho}_n + \Phi_{n+1} \mathbf{x}_{n+1} p_{n+1} = \mathbf{h}_n - \mathbf{k}_{n+1} \mathbf{x}_{n+1}^\top \mathbf{h}_n + \Phi_{n+1} \mathbf{x}_{n+1} p_{n+1}. \quad (2.23)$$

5.  $(\mathbf{A} + \mathbf{BCD})^{-1} = \mathbf{A}^{-1} - \mathbf{A}^{-1} \mathbf{B} (\mathbf{C}^{-1} + \mathbf{DA}^{-1} \mathbf{B})^{-1} \mathbf{DA}^{-1}$ , soit

$$(\mathbf{R}_{n+1})^{-1} = (\mathbf{R}_n)^{-1} - \frac{(\mathbf{R}_n)^{-1} \mathbf{x}_{n+1} \mathbf{x}_{n+1}^\top (\mathbf{R}_n)^{-1}}{1 + \mathbf{x}_{n+1}^\top (\mathbf{R}_n)^{-1} \mathbf{x}_{n+1}}$$

Comme  $\mathbf{k}_{n+1} = \Phi_{n+1}\mathbf{x}_{n+1}$ , nous trouvons la récurrence suivante sur l'équation de Wiener :

$$\mathbf{h}_{n+1} = \mathbf{h}_n + \mathbf{k}_{n+1}(p_{n+1} - \mathbf{h}_n^\top \mathbf{x}_{n+1}). \quad (2.24)$$

Enfin, en notant  $e_{n+1}$  l'erreur *a priori* à l'instant  $n + 1$  comme :

$$e_{n+1} = p_{n+1} - \mathbf{h}_n^\top \mathbf{x}_{n+1}, \quad (2.25)$$

la récurrence s'écrit finalement :

$$\mathbf{h}_{n+1} = \mathbf{h}_n + \mathbf{k}_{n+1}e_{n+1}. \quad (2.26)$$

Pour coder le filtre RLS à l'instant  $n + 1$ , il faut donc exécuter dans l'ordre les équations (2.25), (2.21), (2.26) et (2.20).

Il est prouvé [25] que le filtre RLS converge vers la solution de Wiener :

$$\lim_{n \rightarrow \infty} \mathbf{h}_n = \mathbf{h}_o. \quad (2.27)$$

Cet algorithme récursif permet d'approcher la solution de Wiener à partir des dernières informations disponibles. De plus, il n'est pas nécessaire de stocker tous les signaux, mais uniquement les  $N$  derniers échantillons du signal d'entrée, le dernier échantillon du signal de sortie, ainsi que le vecteur  $\mathbf{h}_n$  (de taille  $N \times 1$ ) et la matrice  $\Phi_n$  (de taille  $N \times N$ ).

Cette version ne permet pas de s'adapter à un système variant dans le temps. En effet, les valeurs mesurées "anciennes" gardent un poids important par rapport aux valeurs mesurées actuelles. Une solution possible pour contourner ce problème est de pondérer les valeurs anciennes, comme le fait l'algorithme W-RLS [25], la fonction de coût à minimiser étant alors :

$$J(\mathbf{h}) = \sum_{k=0}^n \lambda^{n-k} e^2(k), \quad (2.28)$$

avec  $\lambda$  le facteur d'oubli.

### 2.3.3 Algorithme *Least Mean Square* (LMS)

Comme tous les filtres adaptatifs, l'algorithme LMS (*Least Mean Square*) utilise une erreur, définie comme la différence entre la mesure  $p(n)$  et un signal estimé  $\hat{p}(n)$ . Plus

précisément, l'algorithme LMS minimise l'erreur quadratique instantanée (approximation de WINDROW [37]), la fonction de coût étant définie comme :

$$J(\mathbf{h}) = e^2(n) = \left(p(n) - \mathbf{h}_n^\top \mathbf{x}_n\right)^2, \quad (2.29)$$

avec  $p(n)$  l'échantillon du signal mesuré à l'instant courant  $n$ ,  $\mathbf{x}_n$  le vecteur ligne contenant les  $N$  derniers échantillons du signal d'entrée à l'instant  $n$ , et  $\mathbf{h}_n$  le vecteur constitué des  $N$  coefficients de filtre FIR.

Le développement de la fonction de coût donne :

$$J(\mathbf{h}) = p^2(n) - 2p(n)\mathbf{h}_n^\top \mathbf{x}_n + \mathbf{x}_n^\top \mathbf{h}_n \mathbf{h}_n^\top \mathbf{x}_n. \quad (2.30)$$

La méthode d'optimisation de l'algorithme LMS est une descente de gradient, chaque itération de la descente de gradient étant effectuée au cours du temps. Nous cherchons donc une expression pour la mise à jour du filtre de la forme :

$$\mathbf{h}_{n+1} = \mathbf{h}_n - \frac{1}{2}\mu \nabla J(\mathbf{h})|_{\mathbf{h}=\mathbf{h}_n}, \quad (2.31)$$

avec  $\mu$  le pas de descente du gradient.

Or, le gradient de la fonction de coût vérifie :

$$\nabla J(\mathbf{h})|_{\mathbf{h}=\mathbf{h}_n} = -2p(n)\mathbf{x}_n + 2\mathbf{x}_n^\top \mathbf{h}_n \mathbf{x}_n = -2e(n)\mathbf{x}_n. \quad (2.32)$$

D'où, en injectant l'équation du gradient dans l'équation de mise à jour :

$$\mathbf{h}_{n+1} = \mathbf{h}_n + \mu e(n)\mathbf{x}_n. \quad (2.33)$$

Le choix de  $\mu$  est critique car un pas trop faible rend l'algorithme très lent et nécessite de longs temps de mesure. De même, un pas trop grand peut faire diverger l'algorithme (qui ne trouvera alors aucune solution). La convergence du filtre LMS a été largement étudiée dans la littérature [25], [37] et une condition théorique sur le pas peut être formulée suivant :

$$0 < \mu < \frac{2}{\lambda_{\max}}, \quad (2.34)$$

avec  $\lambda_{\max}$  la plus grande valeur propre de la matrice d'autocorrélation du signal d'entrée.

Dans le cas classique d'un signal d'excitation blanc, gaussien, centré et de variance  $\sigma_x^2$ , une condition suffisante<sup>6</sup> de convergence est :

$$0 < \mu < \frac{2}{N\sigma_x^2}. \quad (2.35)$$

Le choix de  $\mu$  fixe également les performances de l'algorithme en termes de valeur finale de l'erreur. En effet, toujours dans le cas où  $x(n) \equiv \mathcal{N}(0, \sigma_x^2)$ , le pas  $\mu$  apparaît dans l'équation de l'erreur finale [25] :

$$J_\infty \approx J_{\min} + \frac{\mu J_{\min}}{2} N\sigma_x^2, \quad (2.36)$$

avec  $J_{\min}$  la valeur minimale de la fonction coût du filtre de Wiener (Eq. 2.14). Donc, plus le pas est petit, plus la solution trouvée par le filtre LMS est proche de la solution optimale de Wiener. Ce pas apparaît également dans le temps de convergence moyen du filtre LMS [25] :

$$\tau_{\text{mse}} = \frac{1}{2\mu\sigma_x^2}, \quad (2.37)$$

ce qui revient à dire que choisir un grand pas permet d'accélérer le temps de convergence du filtre.

Le choix du pas fixe  $\mu$  impose donc les performances de l'algorithme, à savoir vitesse et erreur finale, en sachant que le choix est mutuellement antithétique : privilégier la vitesse de convergence s'effectue au détriment de la précision et réciproquement.

### 2.3.4 Algorithme NLMS

Le filtre LMS est fortement dépendant des propriétés statistiques du signal d'entrée : le choix du pas  $\mu$  dépend de la variance du signal d'entrée et les performances de l'algorithme font également apparaître la variance du signal d'entrée. Dans le cas où le signal d'entrée est inconnu, voire dans le cas où ses propriétés statistiques varient au cours du temps, le choix du paramètre  $\mu$  semble par conséquent très délicat voire impossible.

---

6. Mais pas optimale au sens de la vitesse de convergence et/ou de l'erreur résiduelle. Cette condition assure uniquement la convergence de l'algorithme. Cette condition est vérifiée car  $\lambda_{\max} > \text{Tr}[\mathbf{R}]$ .

Une solution consiste alors à normaliser le paramètre  $\mu$  par le produit  $\mathbf{x}_n \mathbf{x}_n^\top$  [25] :

$$\mu(n) = \frac{\tilde{\mu}}{\mathbf{x}_n \mathbf{x}_n^\top}, \quad (2.38)$$

avec  $\mathbf{x}_n \mathbf{x}_n^\top$  correspondant à la norme euclidienne au carré du signal d'entrée calculée sur les  $N$  derniers échantillons et  $0 < \epsilon \ll 1$  un petit paramètre permettant d'éviter une division par zéro.

L'algorithme NLMS peut être vu comme un filtre LMS à pas variable. L'équation de mise à jour (2.33) devient alors :

$$\mathbf{h}_{n+1} = \mathbf{h}_n + \frac{\tilde{\mu}}{\mathbf{x}_n \mathbf{x}_n^\top + \epsilon} e(n) \mathbf{x}_n. \quad (2.39)$$

Dans ce cas, le paramètre  $\tilde{\mu}$  devient sans dimension et peut être choisi tel que :

$$0 < \tilde{\mu} < 2. \quad (2.40)$$

Cette variante de l'algorithme LMS, appelée NLMS (pour *Normalized Least Mean Square*), permet de s'affranchir de la connaissance *a priori* de la variance du signal d'entrée, et puisqu'elle est estimée à chaque instant, elle laisse la possibilité à l'algorithme de s'adapter en cas de changement de variance du signal d'entrée au cours du temps.

Par contre, il est à noter que les performances de l'algorithme dépendent toujours du choix de ce paramètre : un petit pas donne une faible erreur résiduelle, mais une lente convergence, et un grand pas donne une erreur résiduelle importante, mais un temps de convergence plus faible.

### 2.3.5 Algorithme VSS-LMS

Pour contourner la problématique du choix *a priori* du paramètre  $\mu$  (ou  $\tilde{\mu}$ ), fixant ainsi les performances de l'algorithme, une stratégie consiste à faire évoluer la valeur de ce paramètre au cours du temps, en fonction notamment de l'erreur.

En effet, choisir au départ une valeur importante du pas permet de converger rapidement vers une estimation, puis, lorsque l'erreur devient plus petite, le pas peut être diminué



pour affiner l'estimation. Cela revient donc à avoir un pas variable  $\mu(n)$ , ce que proposent les méthodes dites VSS (*Variable Step Size*). Si le choix de la méthode de calcul de ce pas variable a fait l'objet de nombreuses études [38], il existe cependant des contraintes communes :

- le pas doit rester positif (sinon cela reviendrait à remonter le gradient, et donc à s'éloigner de la solution) ;
- la borne maximale doit être respectée (la borne étant définie suivant l'algorithme de base : LMS ou NLMS), sinon le filtre diverge ;
- il faut éviter la situation limite  $\lim_{n \rightarrow \infty} \mu(n) = 0$ , car le filtre perdrait toute capacité d'adaptation.

De manière générale, les algorithmes VSS-LMS se mettent sous la forme :

$$\mathbf{h}_{n+1} = \mathbf{h}_n + \mu(n)e(n)\mathbf{x}_n. \quad (2.41)$$

Par exemple, SHAN *et al.* [39] proposent un algorithme à pas variable qui tend à diminuer le pas lorsque l'erreur diminue. L'équation de mise à jour du pas est :

$$\mu(n) = \frac{\alpha |\rho(n)|}{\mathbf{x}_n \mathbf{x}_n^\top}, \quad (2.42)$$

la quantité  $\rho(n)$  étant une moyenne des gradients passés :

$$\rho(n) = \lambda \rho(n-1) + (1-\lambda)x(n)e(n). \quad (2.43)$$

Deux paramètres à choisir *a priori* apparaissent dans l'algorithme :  $\lambda$  et  $\alpha$ .

Dans un contexte non stationnaire, le choix du pas est d'autant plus crucial, car il a un impact sur la vitesse de convergence, l'erreur résiduelle, ainsi que sur la capacité de suivi des variations et la stabilité de l'algorithme.

KARNI *et al.* proposent une mise à jour du pas pour un contexte non stationnaire de la forme :

$$\mu(n) = \mu_{\max} \left( 1 - e^{\alpha \|\mathbf{x}_n e(n)\|^2} \right), \quad (2.44)$$

avec  $\alpha$  un paramètre qui pilote la dynamique du filtre et  $\mu_{\max}$  un paramètre qui assure la stabilité.

Le principal problème de ce type d'algorithme est la présence de nombreux paramètres à régler finement. Ces algorithmes sont ainsi souvent écrits pour résoudre un problème spécifique et utilisent des propriétés particulières (RI piquées, courtes, etc.) pour accélérer et/ou améliorer l'erreur résiduelle.

### 2.3.6 Algorithme de méta-gradient

Le filtre LMS et ses déclinaisons souffrent d'une grande sensibilité à la variation des valeurs propres de la matrice d'autocorrélation du signal d'entrée. En effet, pour un signal d'excitation quelconque, la constante de temps du coefficient  $k$  est [25] :

$$\tau_k = \frac{1}{2\mu\lambda_k}. \quad (2.45)$$

Ainsi, pour des signaux d'entrée quelconques, il peut y avoir des temps très différents pour l'estimation de chaque coefficient du filtre.

Une nouvelle famille d'algorithmes adaptatifs, déclinées du filtre LMS (au sens où la méthode d'optimisation est toujours une descente de gradient), met à jour non plus un seul paramètre scalaire  $\mu$  (ou  $\tilde{\mu}$ ), mais un vecteur  $\boldsymbol{\mu}$  contenant autant de paramètres que de points dans le filtre  $\mathbf{h}$ . Ainsi, la mise à jour du filtre est de la forme :

$$\mathbf{h}_{n+1} = \mathbf{h}_n + \boldsymbol{\mu}_{n+1} \circ e(n)\mathbf{x}_n, \quad (2.46)$$

où  $\circ$  désigne le produit de Hadamard (produit terme à terme).

Les algorithmes de méta-gradient restent des algorithmes de descente de gradient sans calcul des dérivées secondes. L'objectif est d'estimer un vecteur pas  $\boldsymbol{\mu}_n$  qui soit un compromis entre temps de convergence et erreur résiduelle. Il est également à noter [26] que le temps de calcul est linéaire en fonction du nombre de points de la réponse impulsionnelle à estimer, contrairement aux algorithmes contenant les dérivées secondes qui ont un coût de calcul quadratique.

Dans un contexte SISO, SUTTON *et al.* [26] proposent une règle de mise à jour du vecteur pas  $\boldsymbol{\mu}_{n+1}$  également par descente de gradient de pas  $\kappa$  :

$$\boldsymbol{\alpha}_{n+1} = \boldsymbol{\alpha}_n + \kappa e(n)\mathbf{x}_n \circ \mathbf{h}_n, \quad (2.47)$$

le vecteur  $\boldsymbol{\mu}_n$  étant défini comme l'exponentielle du vecteur  $\boldsymbol{\alpha}_n$  :

$$\boldsymbol{\mu}_n = e^{\boldsymbol{\alpha}_n} \quad (2.48)$$

et  $\mathbf{h}_n$  étant un vecteur mémoire des changements récents :

$$\mathbf{h}_{n+1} = \mathbf{h}_n \left(1 - \boldsymbol{\mu}_{n+1} \circ \mathbf{x}_n^2\right)^+ + \boldsymbol{\mu}_{n+1} \circ e(n)\mathbf{x}_n, \quad (2.49)$$

avec  $(\cdot)^+$  la valeur si elle est positive<sup>7</sup>.

Cet algorithme, appelé Incremental Delta-Bar-Delta (IDBD), a donc un pas variant dans le temps, qui s'adapte à la fois aux changements liés aux caractéristiques des signaux et à un contexte variant dans le temps. Il est piloté par un seul paramètre,  $\kappa$ , qui reste cependant dépendant du problème physique à estimer. De plus, dans un cas variant dans le temps, certains coefficients de la réponse impulsionnelle sont à corriger très rapidement, alors que d'autres coefficients varient au contraire très peu dans le temps. C'est par exemple le cas du son direct. La singularité correspondant au son direct dans une position donnée source-récepteur doit être mise à zéro l'instant d'après, et vice versa pour le coefficient correspondant au son direct dans la position source-récepteur qui suit. Dans le même temps, les coefficients de la réponse impulsionnelle situés en queue de réverbération ont moins de correction à subir. Cette capacité à modifier les pas individuellement permet alors d'accélérer la convergence pour les coefficients à corriger rapidement (car initialement très faux), sans pour autant dégrader la qualité d'estimation des coefficients déjà corrects.

MAHMOOD *et al.* [27] proposent après SUTTON *et al.* une extension de cet algorithme pour rendre l'algorithme IDBD encore moins sensible à son méta-paramètre  $\kappa$ . Ainsi, l'algorithme proposé, appelé Autostep, devient complètement indépendant du système à estimer. C'est cet algorithme, étendu au cas MISO, qui fait l'objet du chapitre 3.

## 2.4 Extension de SISO à MISO

Les algorithmes présentés précédemment sont écrits pour des systèmes une entrée une sortie (SISO). Dans cette section, nous présentons une méthode pour étendre ces algo-

---

7.  $a^+ = a$  si  $a > 0$ , 0 sinon.

ritmes vers un système multi-entrées une sortie (MISO).

Dans le cas MISO (plusieurs haut-parleurs et un seul microphone), l'équation qui régit le système (linéaire) devient :

$$p(n) = \sum_{l=1}^L (h^{(l)} * x^{(l)})(n), \quad (2.50)$$

avec  $x^{(l)}$  le signal émis par le  $l$ -ième haut-parleur parmi les  $L$  et  $h^{(l)}$  la réponse impulsionnelle entre le haut-parleur  $l$  et le microphone.

Les réponses impulsionnelles étant supposées finies de  $N$  coefficients, nous avons alors :

$$p(n) = \sum_{l=1}^L \mathbf{h}^{(l)} \mathbf{x}_n^{(l)\top} = \begin{bmatrix} \mathbf{h}^{(1)} & \dots & \mathbf{h}^{(L)} \end{bmatrix} \cdot \begin{bmatrix} \mathbf{x}_n^{(1)\top} \\ \vdots \\ \mathbf{x}_n^{(L)\top} \end{bmatrix}. \quad (2.51)$$

Une possibilité pour étendre l'équation du filtre LMS-SISO (2.33) au cas MISO est donc de travailler sur un vecteur "réponses impulsionnelles" qui est la concaténation des réponses impulsionnelles :

$$\begin{bmatrix} \mathbf{h}_{n+1}^{(1)\top} \\ \vdots \\ \mathbf{h}_{n+1}^{(L)\top} \end{bmatrix} = \begin{bmatrix} \mathbf{h}_n^{(1)\top} \\ \vdots \\ \mathbf{h}_n^{(L)\top} \end{bmatrix} + \mu e(n) \begin{bmatrix} \mathbf{x}_n^{(1)\top} \\ \vdots \\ \mathbf{x}_n^{(L)\top} \end{bmatrix}, \quad (2.52)$$

avec

$$e(n) = p(n) - \begin{bmatrix} \mathbf{h}^{(1)} & \dots & \mathbf{h}^{(L)} \end{bmatrix} \cdot \begin{bmatrix} \mathbf{x}_n^{(1)\top} \\ \vdots \\ \mathbf{x}_n^{(L)\top} \end{bmatrix}. \quad (2.53)$$

Il est aussi possible de traiter chaque réponse impulsionnelle indépendamment (seule l'erreur est commune à toutes les équations), puisque,  $\forall l \in [1, L]$  :

$$\mathbf{h}_{n+1}^{(l)} = \mathbf{h}_n^{(l)} + \mu e(n) \mathbf{x}_n^{(l)}. \quad (2.54)$$

**Étude de convergence :** Les équations étant toutes liées par l'erreur (unique), une étude de convergence est nécessaire pour valider cette approche.

Définissons le vecteur erreur à l'instant  $n + 1$  comme la différence entre la réponse impulsionnelle réelle  $\mathbf{h}_r^{(l)}$  et son estimation  $\mathbf{h}_{n+1}^{(l)}$  :

$$\boldsymbol{\epsilon}_{n+1}^{(l)} = \mathbf{h}_r^{(l)} - \mathbf{h}_{n+1}^{(l)}. \quad (2.55)$$

En injectant l'équation de mise à jour du filtre, nous obtenons :

$$\boldsymbol{\epsilon}_{n+1}^{(l)} = \mathbf{h}_r^{(l)} - \mathbf{h}_n^{(l)} - \mu^{(l)} e(n) \mathbf{x}_n^{(l)}. \quad (2.56)$$

Remplaçons alors l'erreur par sa définition :

$$\boldsymbol{\epsilon}_{n+1}^{(l)} = \mathbf{h}_r^{(l)} - \mathbf{h}_n^{(l)} - \mu^{(l)} \left( p(n) - \sum_{j=1}^L \mathbf{h}_n^{(j)} \mathbf{x}_n^{(j)\top} \right) \mathbf{x}_n^{(l)}. \quad (2.57)$$

Or le signal mesuré  $p(n)$  est la convolution de la réponse impulsionnelle réelle avec le signal d'entrée à laquelle s'ajoute un bruit de mesure  $b(n)$  (voir figure 2.1), donc :

$$\boldsymbol{\epsilon}_{n+1}^{(l)} = \mathbf{h}_r^{(l)} - \mathbf{h}_n^{(l)} - \mu^{(l)} \left[ \sum_{j=1}^L (\mathbf{h}_r^{(j)} - \mathbf{h}_n^{(j)}) \mathbf{x}_n^{(j)\top} \right] \mathbf{x}_n^{(l)} \quad (2.58)$$

$$- \mu^{(l)} b(n) \mathbf{x}_n^{(l)}. \quad (2.59)$$

Après réarrangement des termes et factorisation par  $\boldsymbol{\epsilon}_n^{(l)}$ , l'erreur s'écrit sous la forme :

$$\begin{aligned} \boldsymbol{\epsilon}_{n+1}^{(l)} &= \left( \mathbf{I} - \mu^{(l)} (\mathbf{x}_n^{(l)} \mathbf{x}_n^{(l)\top}) \right) \boldsymbol{\epsilon}_n^{(l)} \\ &\quad - \sum_{\substack{j=1 \\ j \neq l}}^L \left( \mu^{(l)} \mathbf{x}_n^{(j)} \mathbf{x}_n^{(l)\top} \boldsymbol{\epsilon}_n^{(j)} \right) - \mu^{(l)} b(n) \mathbf{x}_n^{(l)}, \end{aligned} \quad (2.60)$$

et son espérance vaut alors :

$$\begin{aligned} \mathbb{E}[\boldsymbol{\epsilon}_{n+1}^{(l)}] &= \left( \mathbf{I} - \mu^{(l)} \mathbb{E} [\mathbf{x}_n^{(l)} \mathbf{x}_n^{(l)\top}] \right) \mathbb{E} [\boldsymbol{\epsilon}_n^{(l)}] \\ &\quad - \sum_{\substack{j=1 \\ j \neq l}}^L \mu^{(l)} \mathbb{E} [\mathbf{x}_n^{(j)} \mathbf{x}_n^{(l)\top}] \mathbb{E} [\boldsymbol{\epsilon}_n^{(j)}] \\ &\quad - \mu^{(l)} \mathbb{E} [b(n) \mathbf{x}_n^{(l)}]. \end{aligned} \quad (2.61)$$

Supposons le bruit de mesure  $b(n)$  décorrélié des signaux d'entrée  $\mathbf{x}_n^{(l)}$ , alors  $\mathbb{E} [b(n)\mathbf{x}_n^{(l)}] = 0$ . De plus, si les signaux d'entrée sont mutuellement décorrélés, les termes croisés  $j \neq l$  sont alors nuls, soit  $\mathbb{E} [\mathbf{x}_n^{(j)}\mathbf{x}_n^{(l)\top}] = 0$ . Nous obtenons finalement l'équation suivante :

$$\mathbb{E}[\boldsymbol{\epsilon}_{n+1}^{(l)}] = (\mathbf{I} - \mu^{(l)}\mathbf{R}^{(l)}) \mathbb{E}[\boldsymbol{\epsilon}_n^{(l)}]. \quad (2.62)$$

Le filtre MISO converge alors dans les mêmes conditions que le filtre SISO, à condition que les signaux d'entrée soient mutuellement décorrélés.

## 2.5 Conclusion

Lors de l'étape de calibration du système, une estimation des réponses impulsionnelles peut être effectuée par mesure classique en considérant  $L \times M$  systèmes SISO. Par contre, lors de l'exploitation du système, les entrées n'étant plus indépendantes, il est nécessaire d'utiliser une estimation MISO. Le filtrage adaptatif permet, en temps réel, d'estimer les réponses impulsionnelles. Cette capacité à s'adapter au cours du temps permet alors de connaître le système même si ses caractéristiques ont changé.

Cependant, les filtres adaptatifs ont souvent de nombreux paramètres à régler, et ces paramètres sont souvent liés aux caractéristiques des signaux et/ou du système.

Dans le cas où le système peut varier au cours du temps, les paramètres judicieux au départ peuvent devenir sous-optimaux, voire dans certains cas amener à des instabilités. La classe d'algorithmes dits méta-gradient offre des performances très intéressantes et surtout repose sur des paramètres "généraux" indépendants du système physique considéré.

Dans notre cas, les réponses impulsionnelles pouvant être très différentes en fonction de la géométrie du système et des caractéristiques de la salle, ces algorithmes permettent ainsi d'estimer les réponses impulsionnelles du système de *sound zones* sans avoir à régler et paramétrer finement les algorithmes.

Cependant, il n'existe pas d'algorithme méta-gradient capable d'estimer des réponses impulsionnelles d'un système MISO. L'objet du prochain chapitre est donc d'étendre en l'adaptant l'algorithme de MAHMOOD *et al.* [27] à un système MISO.

# ALGORITHME MISO-AUTOSTEP

---

Dans ce chapitre, l'algorithme Autostep de MAHMOOD est étendu au cas MISO pour l'estimation de réponses impulsionnelles. Ce chapitre a fait l'objet d'un article publié dans le *Journal of the Acoustical Society of America (JASA)* [40]. Dans une première partie, le contexte théorique est précisé : notations, algorithme, convergence. Ensuite, les métriques utilisées par la suite sont détaillées (désalignement, erreur quadratique moyenne, erreur spectrale quadratique moyenne) et le choix des méta-paramètres est analysé au travers d'une première simulation. Les algorithmes MISO-LMS et MISO-Autostep sont comparés dans une troisième partie dans un contexte de simulation et dans une dernière partie dans un contexte expérimental.

## 3.1 Développement théorique

Dans cette partie, nous nous plaçons dans le contexte d'un système MISO invariant dans le temps (figure 3.1).

Les entrées sont notées  $x^{(l)}(n)$  avec  $l \in [1, L]$  le numéro de haut-parleur. La sortie physique mesurée est notée  $p(n)$  et son estimation est notée  $\hat{p}(n)$ . La réponse impulsionnelle réelle entre le microphone et le haut-parleur  $l$  est noté  $h_r^{(l)}(n)$  et son estimation  $h^{(l)}(n)$ . Le signal mesuré est supposé bruité par un bruit additif noté  $b(n)$  et indépendant des entrées. Enfin, nous définissons l'erreur  $e(n)$  comme la différence entre le signal mesuré  $p(n)$  et l'estimation  $\hat{p}(n)$  :

$$e(n) = p(n) - \hat{p}(n) = p(n) - \sum_{l=1}^L \mathbf{h}^{(l)\top} \mathbf{x}_n. \quad (3.1)$$

Dans un contexte MISO, les entrées peuvent avoir des propriétés statistiques différentes. Dans le cas du filtre LMS, les valeurs de pas  $\mu^{(l)}$  doivent alors être adaptées en fonction des *a priori* disponibles. Une autre solution consiste à mettre à jour ces valeurs de pas

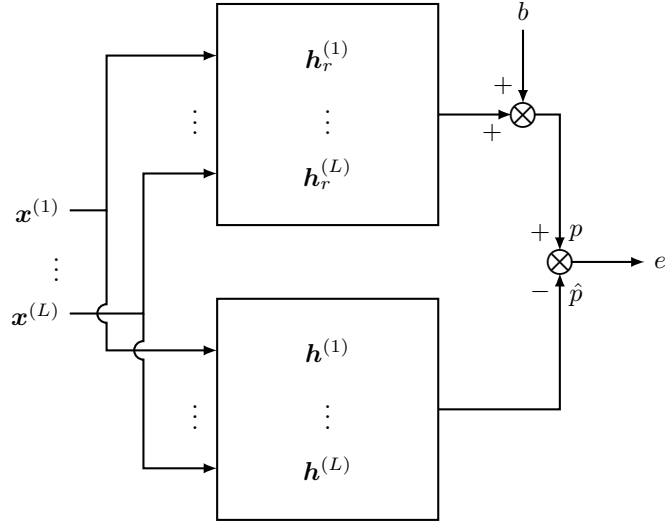


FIGURE 3.1 – Schéma bloc du système MISO.

avec une loi elle-même adaptée. C'est ce qu'a proposé SUTTON dans son algorithme IDBD [26] pour l'identification de système SISO (voir section 2.3.6). Suivant le raisonnement de SUTTON, nous proposons ici une version MISO de cet algorithme. Considérons un vecteur  $N \times 1$  de paramètres  $\boldsymbol{\mu}^{(l)}$  :

$$\boldsymbol{\mu}^{(l)} = [\mu_0^{(l)}, \mu_1^{(l)}, \dots, \mu_{N-1}^{(l)}]^\top. \quad (3.2)$$

L'équation de mise à jour de  $\mathbf{h}^{(l)}$  au temps  $n + 1$  devient alors

$$\mathbf{h}_{n+1}^{(l)} = \mathbf{h}_n^{(l)} + \boldsymbol{\mu}_{n+1}^{(l)} \circ e_n \mathbf{x}_n^{(l)}, \quad (3.3)$$

où  $\boldsymbol{\mu}_{n+1}^{(l)}$  est adapté incrémentalement par descente de gradient et où  $\circ$  désigne le produit de Hadamard (produit terme à terme). En utilisant la méthode de la référence [26], nous écrivons  $\boldsymbol{\mu}_{n+1}^{(l)}$  sous une forme qui permet d'assurer sa positivité :

$$\boldsymbol{\mu}_{n+1}^{(l)} = \exp[\boldsymbol{\alpha}_{n+1}^{(l)}], \quad (3.4)$$

où  $\boldsymbol{\alpha}^{(l)}$  est un vecteur mis à jour suivant :

$$\boldsymbol{\alpha}_{n+1}^{(l)} = \boldsymbol{\alpha}_n^{(l)} - \kappa \frac{\partial e_n^2}{\partial \boldsymbol{\alpha}^{(l)}}, \quad (3.5)$$

avec  $\kappa > 0$  un méta-paramètre scalaire jouant le rôle du pas de l'algorithme IDBD.



L'algorithme IDBD [26] estime ainsi le meilleur vecteur de pas de descente de gradient, en ajoutant un nouveau paramètre  $\kappa$  dépendant lui aussi du problème physique.

MAHMOOD *et al.*, ont proposé d'exprimer le terme de gradient de l'équation (3.5) sous une forme matricielle  $\mathbf{G}_n^{(l)}$  qui est approximée par une matrice diagonale,

$$\boldsymbol{\alpha}_{n+1}^{(l)} = \boldsymbol{\alpha}_n^{(l)} + \kappa e_n \mathbf{x}_n^{(l)} \circ \mathbf{g}_n^{(l)}, \quad (3.6)$$

où  $\mathbf{g}_n^{(l)}$  est le vecteur  $N \times 1$  contenant les valeurs de la diagonale de la matrice  $\mathbf{G}_n^{(l)}$ . Ces valeurs sont données par [27] :

$$\mathbf{g}_{n+1}^{(l)} = \mathbf{g}_n^{(l)} - \mathbf{g}_n^{(l)} \circ \boldsymbol{\mu}_{n+1}^{(l)} \circ \mathbf{x}_n^{(l)} \circ \mathbf{x}_n^{(l)} + \boldsymbol{\mu}_{n+1}^{(l)} \circ e_n \mathbf{x}_n^{(l)}. \quad (3.7)$$

En insérant l'équation (3.6) dans l'équation (3.4) nous obtenons :

$$\boldsymbol{\mu}_{n+1}^{(l)} = \boldsymbol{\mu}_n^{(l)} \exp[\kappa e_n \mathbf{x}_n^{(l)} \circ \mathbf{g}_n^{(l)}]. \quad (3.8)$$

À l'instant  $n + 1$ , l'algorithme IDBD calcule successivement les équations (3.6), (3.8) et (3.3).

Le méta-paramètre  $\kappa$  restant jusqu'à présent dépendant du problème physique, MAHMOOD *et al.* [27] propose une nouvelle version de l'algorithme IDBD, appliquée aux problèmes SISO et appelée Autostep (voir section 2.3.6) permettant de contourner cette limitation. Pour rendre adimensionnel le paramètre  $\kappa$ , introduisons un nouveau vecteur  $N \times 1$  appelé vecteur de normalisation  $\mathbf{v}$  avec comme règle de mise à jour [27] :

$$\mathbf{v}_{n+1}^{(l)} = \mathbf{v}_n^{(l)} + \gamma \boldsymbol{\mu}_n^{(l)} \circ \mathbf{x}_n^2 \circ \left( \left| \mathbf{x}_n e_n \circ \boldsymbol{\alpha}_n^{(l)} \right| - \mathbf{v}_n^{(l)} \right), \quad (3.9)$$

où  $\gamma$  est un facteur d'oubli. L'équation (3.8) devient alors :

$$\boldsymbol{\mu}_{n+1}^{(l)} = \boldsymbol{\mu}_n^{(l)} \exp[\kappa e_n \mathbf{x}_n^{(l)} \circ \mathbf{g}_n^{(l)} \oslash \mathbf{v}_{n+1}^{(l)}] \quad (3.10)$$

avec  $\oslash$  la division terme à terme.

Pour assurer la stabilité de l'algorithme Autostep avec des signaux réels, il est nécessaire de réaliser quelques modifications de l'équation de mise à jour (3.10). Premièrement, avec des signaux réels, les valeurs des signaux d'entrée  $x$  peuvent varier dans de grandes propor-

tions, alors que la normalisation ne prend pas en compte immédiatement ces variations. Dans ce cas, nous imposons à  $\mathbf{v}_{n+1}^{(l)}$  de vérifier :

$$\mathbf{v}_{n+1}^{(l)} = \max(\mathbf{v}_{n+1}^{(l)}, |e_n \mathbf{x}_n^{(l)} \circ \mathbf{g}_n^{(l)}|). \quad (3.11)$$

Ensuite, l'ajout du vecteur de normalisation  $\mathbf{v}^{(l)}$  au dénominateur peut poser des problèmes de stabilité. En particulier, il est nécessaire d'éviter les divisions par zéro. MAHMOOD *et al.* [27] propose ainsi de remplacer toutes les entrées nulles du vecteur  $\mathbf{v}^{(l)}$  par 1 :

$$\mathbf{v}_{n+1}^{(l)} = 1 \text{ si } \mathbf{v}_{n+1}^{(l)} = 0. \quad (3.12)$$

Enfin, une condition suffisante sur le pas de descente de gradient pour la stabilité est un produit scalaire positif entre deux instants successifs [27]. Dans notre cas, cette condition devient :

$$\left( \frac{\partial \left( p_n - \mathbf{h}_{n+1}^{(l)\top} \mathbf{x}_{n+1}^{(l)} \right)^2}{\partial \mathbf{h}_{n+1}^{(l)}} \right)^\top \left( \frac{\partial \left( p_n - \mathbf{h}_n^{(l)\top} \mathbf{x}_n^{(l)} \right)^2}{\partial \mathbf{h}_n^{(l)}} \right) \geq 0. \quad (3.13)$$

Après dérivation, nous obtenons alors :

$$\left( \left( p_n - \mathbf{h}_{n+1}^{(l)\top} \mathbf{x}_{n+1}^{(l)} \right) \mathbf{x}_{n+1}^{(l)} \right)^\top \left( \left( p_n - \mathbf{h}_n^{(l)\top} \mathbf{x}_n^{(l)} \right) \mathbf{x}_n^{(l)} \right) \geq 0, \quad (3.14)$$

ou de manière équivalente,

$$\left( \boldsymbol{\mu}_{n+1}^{(l)} \circ \mathbf{x}_n^{(l)} \right)^\top \mathbf{x}_n^{(l)} \leq 1. \quad (3.15)$$

Pour éviter la division par 0, l'équation (3.8) est modifiée suivant :

$$\boldsymbol{\mu}_{n+1}^{(l)} = \begin{cases} \log[\boldsymbol{\mu}_{n+1}^{(l)}] & \text{if } \mathbf{v}_{n+1}^{(l)} = 0 \\ \log[\boldsymbol{\mu}_{n+1}^{(l)}] + \kappa e_n \mathbf{x}_n^{(l)} \circ \mathbf{g}_n^{(l)} \oslash \mathbf{v}_{n+1}^{(l)} & \text{else} \end{cases}, \quad (3.16)$$

et si  $(\boldsymbol{\mu}_{n+1}^{(l)} \circ \mathbf{x}_n^{(l)})^\top \mathbf{x}_n^{(l)} > 1$  :

$$\boldsymbol{\mu}_{n+1}^{(l)} = \frac{\boldsymbol{\mu}_{n+1}^{(l)}}{(\boldsymbol{\mu}_{n+1}^{(l)} \circ \mathbf{x}_n^{(l)})^\top \mathbf{x}_n^{(l)}}. \quad (3.17)$$

Le pseudo code de l'algorithme MISO-Autostep est résumé dans **Algorithme 1** (page

35).

Les vecteurs  $\mathbf{v}$  et  $\boldsymbol{\alpha}$  sont initialisés à zéro. Le vecteur contenant les réponses impulsionnelles  $\mathbf{h}$  peut être initialisé à zéro, cependant s'il y a des *a priori* disponibles sur les réponses impulsionnelles, il est possible de les intégrer à l'initialisation. En effet, plus le vecteur  $\mathbf{h}$  à l'initialisation est proche de la solution, plus la convergence sera rapide.

Le vecteur pas  $\mu$  peut être initialisé à zéro, cependant pour accélérer la convergence, il est possible de l'initialiser autour d'une valeur qui assure la convergence comme la valeur que l'on donnerait à un filtre LMS.<sup>1</sup>

Les deux méta-paramètres sont  $\kappa$  et  $\gamma$ , les valeurs à choisir font l'objet d'une étude dans la section 3.2.2.

L'algorithme nous retourne les réponses impulsionnelles estimées à l'instant  $n$ , ainsi que l'erreur résiduelle à l'instant  $n$ .

**Note sur la convergence** Comme il a été montré dans la section 2.4, l'algorithme MISO converge avec les mêmes propriétés que l'algorithme SISO du départ, à condition que les signaux d'entrées soient mutuellement décorrélés. Le détail de calcul propre à l'algorithme MISO-Autostep est proposé en Annexe A.

---

1. Il a été testé des valeurs très grandes d'initialisation de ce vecteur, qui dépassent la valeur qui assure la convergence (Eq A.14), sans que pour autant l'algorithme ne diverge complètement. Par contre, comme au départ, il s'écarte fortement de la valeur optimale, les temps de convergence explosent.

---

**Algorithme 1 : MISO-Autostep**


---

**Result :  $\mathbf{h}$**

- 1  $\mathbf{h} = \mathbf{0}$ ;
- 2  $\mathbf{v} = \mathbf{0}$ ;
- 3  $\boldsymbol{\alpha} = \mathbf{0}$ ;
- 4  $\boldsymbol{\mu} = \frac{1}{NL\sigma_{x^{(l)}}^2}$ ;
- 5  $\kappa = 1 \times 10^{-2}$ ;
- 6  $\gamma = 1 \times 10^{-4}$ ;
- 7 **while**  $\{x_n, p_n\}$  **do**
- 8      $e_n = p_n - \sum_{l=1}^L \mathbf{h}_n^{(l)\top} \mathbf{x}_n^{(l)}$ ;
- 9      $\mathbf{v}_{n+1}^{(l)} = \mathbf{v}_n^{(l)} + \gamma \boldsymbol{\mu}_n^{(l)} \circ \mathbf{x}_n^{(l)2} \circ \left( \left| \mathbf{x}_n^{(l)} \circ e_n \boldsymbol{\alpha}_n^{(l)} \right| - \mathbf{v}_n^{(l)} \right)$ ;
- 10    **if**  $\mathbf{v}_{n+1}^{(l)} = \mathbf{0}$  **then**
- 11     |  $\boldsymbol{\alpha}_n^{(l)} = \log\{\boldsymbol{\mu}_n^{(l)}\}$ ;
- 12    **else**
- 13     |  $\boldsymbol{\alpha}_n^{(l)} = \log\{\boldsymbol{\mu}_n^{(l)}\} + \kappa e_n \mathbf{x}_n^{(l)} \circ \mathbf{g}^{(l)} \oslash \mathbf{v}_{n+1}^{(l)}$ ;
- 14    **end**
- 15     $\boldsymbol{\mu}_n^{(l)} = \exp \boldsymbol{\alpha}_n^{(l)}$ ;
- 16     $\boldsymbol{\mu}_n^{(l)} = \boldsymbol{\mu}_n^{(l)} \oslash \max\{\boldsymbol{\mu}_n^{(l)} \circ \mathbf{x}_n^{(l)2}, 1\}$ ;
- 17     $\mathbf{h}_{n+1}^{(l)} = \mathbf{h}_n^{(l)} + \boldsymbol{\mu}_n^{(l)} \circ e_n \mathbf{x}_n^{(l)}$ ;
- 18     $\mathbf{g}_{n+1}^{(l)} = (1 - \boldsymbol{\mu}_n^{(l)} \circ \mathbf{x}_n^{(l)2}) \mathbf{g}_n^{(l)} + \boldsymbol{\mu}_n^{(l)} \circ e_n \mathbf{x}_n^{(l)}$ ;
- 19 **end**

---

## 3.2 Métriques et optimisation préliminaire

### 3.2.1 Métriques

Pour pouvoir analyser les performances des algorithmes et les comparer, nous définissons ici les métriques utilisées par la suite.

**Désalignement** Le désalignement (*misalignment* en anglais) code la distance euclidienne entre deux vecteurs. Notons  $\mathbf{h}^{(l)}$  la réponse impulsionnelle réelle de l'entrée  $l$  et  $\hat{\mathbf{h}}^{(l)}(n)$  son estimation à l'instant  $n$ , le *misalignment* est défini par :

$$\text{mis}^{(l)}(n) = \|\mathbf{h}^{(l)} - \hat{\mathbf{h}}_n^{(l)}\|_2, \quad (3.18)$$

avec  $\|\cdot\|_2$  la norme  $L_2$ . Pour une simulation de  $C$  cycles d'adaptation, nous notons  $\text{mis}^{(l)} \equiv \text{mis}^{(l)}(C)$  et  $\langle \text{mis} \rangle$  sa moyenne sur le nombre total de haut-parleurs.

**Erreur quadratique moyenne** À partir du signal d'erreur  $e_n$  défini par l'équation (3.1), l'erreur quadratique moyenne (MSE) est définie comme l'espérance du carré de l'erreur :

$$\text{mse}(n) = \mathbb{E} [e_n^2]. \quad (3.19)$$

Pour une seule réalisation, l'erreur quadratique moyenne est approximée par

$$\text{mse}(n) = \frac{1}{K} \sum_{i=n-K+1}^n e_i^2, \quad (3.20)$$

avec  $K$  un entier positif.

**Erreur spectrale quadratique moyenne** Pour certaines applications, les performances doivent être estimées dans une bande fréquentielle d'intérêt. Notons  $H^{(l)}(f)$  la transformée de Fourier discrète (TFD) de la réponse impulsionnelle réelle (i.e. la FRF) et  $\hat{H}^{(l)}(f)$  la TFD de la réponse impulsionnelle estimée. Nous définissons l'erreur spectrale quadratique moyenne  $\text{mse}_{\text{DFT}}^{(l)}$  comme :

$$\text{mse}_{\text{DFT}}^{(l)} = \sum_{f=f_1}^{f_2} \frac{\left( \left| \hat{H}^{(l)}(f) \right| - \left| H^{(l)}(f) \right| \right)^2}{\left| H^{(l)}(f) \right|^2}, \quad (3.21)$$

et notons  $\langle \text{mse}_{\text{DFT}} \rangle$  sa moyenne sur tous les haut-parleurs.

### 3.2.2 Choix des méta-paramètres

L'algorithme Autostep a été initialement créé par MAHMOOD pour rendre l'algorithme IDBD de SUTTON indépendant du problème physique auquel il s'applique. MAHMOOD l'a testé sur divers problèmes avec succès sans changer la valeur des méta-paramètres par défaut [27]. D'autres auteurs l'ont également utilisé sur d'autres problématiques sans changer la valeur des méta-paramètres [41], [42]. Pour valider le choix des méta-paramètres dans un contexte d'estimation de réponses impulsionnelles d'un système MISO, nous avons choisi un scénario simple (points sources rayonnant en espace libre) et nous avons testé l'algorithme à l'aide d'une méthode bayésienne.

Pour chaque entrée (haut-parleur)  $l$ , définissons la fonction coût basée sur le désalignement :

$$J^{(l)}(\kappa, \gamma) = \mathbb{E} [\text{mis}^{(l)}]^2 + \text{Var} [\text{mis}^{(l)}], \quad (3.22)$$

Var désignant la variance, et considérons la moyenne  $J(\kappa, \gamma)$  des  $L$  fonctions coût, soit :

$$J(\kappa, \gamma) = \frac{1}{L} \sum_{l=1}^L J^{(l)}(\kappa, \gamma). \quad (3.23)$$

Le choix de la fonction  $J(\kappa, \gamma)$  permet de trouver un compromis entre une faible erreur d'estimation de chaque réponse impulsionnelle et éviter trop de variations entre les estimations de chaque entrée.

Pour illustrer le rôle des paramètres  $\kappa$  et  $\gamma$  dans le processus d'optimisation, une propagation acoustique en champ libre est simulée ici, en considérant quatre sources (haut-parleurs)<sup>2</sup> et un microphone, la distance entre chaque haut-parleur et le microphone étant identique et l'objectif étant d'estimer les quatre réponses impulsionnelles haut-parleur/microphone. Les réponses impulsionnelles peuvent être modélisées analytiquement par [43] :

$$h^{(l)}(n) = \frac{1}{4\pi d^{(l)}} \delta \left( n - \left\lfloor \frac{d^{(l)}}{c_0 \cdot T_s} \right\rfloor \right), \quad (3.24)$$

---

2. Les haut-parleurs sont ici considérés comme des monopoles.

avec  $d^{(l)}$  la distance entre le haut-parleur  $l$  et le microphone,  $T_s$  la période d'échantillonnage,  $\delta(n)$  l'impulsion discrète,  $c_0$  la célérité du son dans l'air et  $\lfloor \alpha \rfloor$  l'arrondi de  $\alpha$  à l'entier inférieur le plus proche.

Pour minimiser la fonction coût précédemment définie, la sélection des méta-paramètres est effectuée sur une grille avec validation croisée où les combinaisons possibles sont sélectionnées avec une approche bayésienne [44]. D'après MAHMOOD *et al.* [27], les valeurs des méta-paramètres peuvent être choisies telles que  $\kappa \in [1 \times 10^{-11}, 1 \times 10^2]$ ,  $\gamma \in [1 \times 10^{-11}, 1 \times 10^2]$ . Les autres paramètres de la simulation sont :  $L = 4$ ,  $d^{(l)} = 1$  m,  $c_0 = 343 \text{ m s}^{-1}$ ,  $F_s = 1/T_s = 16 \text{ kHz}$  et  $N = 100$  points. La durée de simulation est  $T = 5$  s, par conséquent  $C = 80000$  itérations. La fonction coût est évaluée sur 200 points. Les quatre réponses impulsionnelles sont ensuite convoluées avec quatre bruits blancs centrés indépendants de variance unitaire et sont finalement sommées. Enfin, un bruit blanc gaussien est ajouté au signal de sortie tel que le rapport signal sur bruit soit de 30 dB.

La figure 3.2 montre l'évolution de la fonction de coût  $J(\kappa, \gamma)$  en fonction de  $\kappa$  et  $\gamma$ . L'algorithme MISO-Autostep présente une très faible sensibilité à ces paramètres, comme illustré par la présence d'un plateau très étendu où  $J(\kappa, \gamma)$  est d'amplitude très faible. Les valeurs  $\kappa$  et  $\gamma$  peuvent donc être choisies dans une grande plage. Ces résultats sont conformes à ceux rapportés dans la littérature [41], [42].

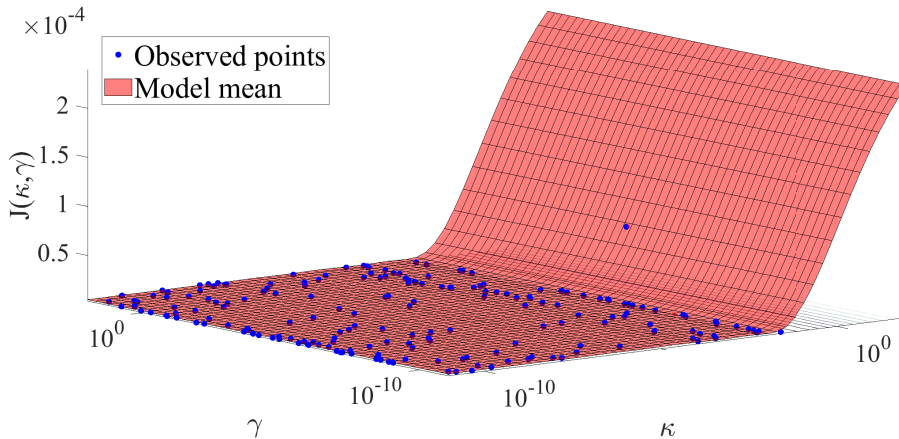


FIGURE 3.2 – Fonction de coût  $J(\kappa, \gamma)$  en fonction des méta-paramètres  $\kappa$  et  $\gamma$ . Les points bleus représentent les calculs de  $J$  tandis que la surface orange représente le modèle de régression du processus gaussien. Pour une large plage de valeurs de  $(\kappa, \gamma)$ ,  $J(\kappa, \gamma) \simeq 5,1 \times 10^{-6}$ . La valeur minimale de  $J(\kappa, \gamma)$  est obtenue pour  $(\kappa, \gamma) = (0,22, 4,7 \times 10^{-10})$ .

Pour vérifier ces résultats et analyser les possibles effets de  $(\kappa, \gamma)$  sur la convergence de l'algorithme, plusieurs courbes de convergence sont estimées par méthode de Monte Carlo (1000 essais). Les vecteurs  $\mathbf{h}$ ,  $\mathbf{v}$ ,  $\boldsymbol{\alpha}$  sont initialisés par un vecteur nul et tous les éléments du vecteur  $\boldsymbol{\mu}$  valent arbitrairement  $1 \times 10^{-4}$  à l'initialisation. Nous considérons dans un premier temps  $\kappa = 0.01$  [27]. La figure 3.3 présente quatre courbes d'apprentissage correspondant à quatre valeurs de  $\gamma$  prises dans une très grande plage de valeurs possibles, montrant que le paramètre  $\gamma$  n'a aucune influence sur la précision de l'algorithme<sup>3</sup>. Nous

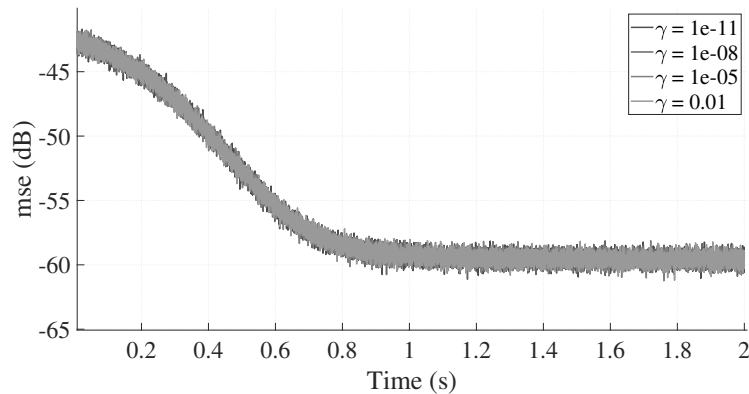


FIGURE 3.3 – *Learning curves* de l'algorithme MISO-Autostep pour différentes valeurs de  $\gamma$  avec  $\kappa = 0.01$ .

poursuivons en fixant  $\gamma = 0.01$  et en analysant l'effet de  $\kappa$  (figure 3.4). Contrairement à ce qui a été observé avec  $\gamma$ , l'algorithme converge plus rapidement lorsque  $\kappa$  possède une grande valeur. Pour rappel, le paramètre  $\kappa$  apparaît dans l'équation de mise à jour de  $\boldsymbol{\mu}$  (3.5) : plus sa valeur est grande, plus la convergence est rapide.

Pour terminer, la figure 3.5 présente différentes courbes d'apprentissage (*learning curves*) pour l'algorithme MISO-LMS pour différents paramètres de pas  $\mu$ . Une comparaison avec la courbe d'apprentissage de MISO-Autostep ( $\kappa = 0.01$  et  $\gamma = 0.01$ ) est proposée. Les courbes MISO-LMS montrent qu'avec un pas grand la convergence est rapide, mais qu'elle atteint une erreur plus grande que lorsque le pas est plus petit. L'algorithme MISO-Autostep présente quant à lui un compromis entre la vitesse de convergence et l'erreur résiduelle finale.

3. Rappelons que nous sommes dans un contexte invariant dans le temps avec des signaux stationnaires. Le paramètre  $\gamma$  étant assimilable à une constante de temps, il peut avoir une influence dans un cas variant dans le temps.



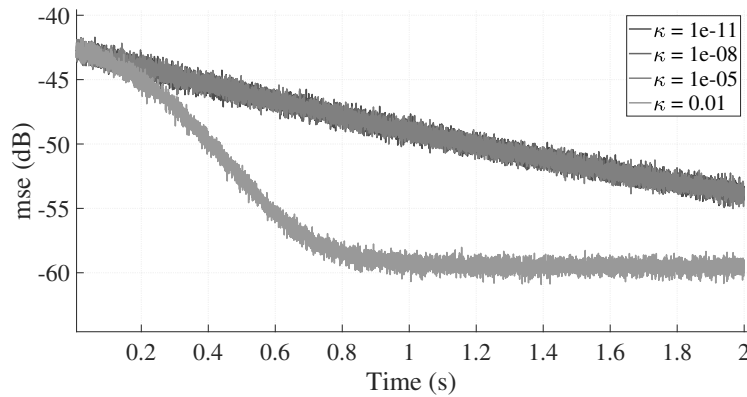


FIGURE 3.4 – *Learning curves* de l’algorithme MISO-Autostep pour différentes valeurs de  $\kappa$  avec  $\gamma = 0.01$ .

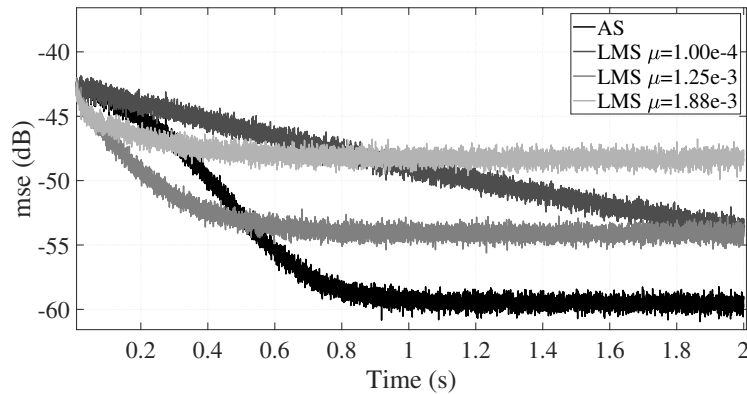


FIGURE 3.5 – *Learning curves* des algorithmes MISO-Autostep et MISO-LMS. Pour MISO-LMS, trois valeurs de  $\mu$  sont présentées. Pour MISO-Autostep (MISO-AS),  $\kappa = 0.01$  et  $\gamma = 0.01$ .

### 3.2.3 Temps de calcul

Lors des simulations précédentes, le temps de calcul pour les algorithmes MISO-LMS, MISO-Autostep et MISO-RLS ont été mesurés pour comparaison. Les temps de calculs dépendent du nombre de coefficients à estimer et sont affichés sur le tableau 3.1.

Le temps de calcul pour l’algorithme LMS est linéaire en fonction du nombre de coefficients à estimer, et quadratique pour l’algorithme RLS, ce qui concorde avec la littérature [25]. L’algorithme MISO-Autostep est trois fois plus gourmand en temps de calcul comparé à l’algorithme LMS, cependant le temps de calcul reste linéaire en fonction du nombre

TABLE 3.1 – Médiane du temps de calcul (sur 100 essais) | Intel i7-10875H @ 2,3 GHz | 64 Gb RAM | Script Matlab

Nombre de coefficients ( $\times 6$ )	Autostep	LMS	RLS
10	0,0023	0,0009	0,0373
20	0,0030	0,0011	0,0363
50	0,0051	0,0016	0,0594
100	0,0085	0,0021	0,1363
200	0,0166	0,0031	0,4094
500	0,0413	0,0069	5,8570
1000	0,0663	0,0125	19,3416
2000	0,1239	0,0219	55,0720
Temps de calcul (s)	$0.0036N$	$0.0011N$	$0.0101N^2 - 0.7155N$

de coefficients à estimer. Sur des réponses impulsionnelles longues, comme c’est le cas en acoustique des salles, l’algorithme MISO-Autostep aura un avantage majeur en terme de temps de calcul par rapport à l’algorithme RLS.

### 3.3 Comparaison des algorithmes MISO-LMS et MISO-Autostep dans un environnement simulé

Dans cette partie, les algorithmes MISO-LMS et MISO-Autostep sont testés dans un contexte d’identification de réponses impulsionnelles de salle, avec plusieurs sources actives en même temps. La pièce simulée ici est un parallélépipède rectangle de dimensions :  $4\text{ m} \times 3\text{ m} \times 2,5\text{ m}$  et possède une durée de réverbération RT60 de 0,5 s.

Six sources monopolaires sont alignées et un microphone est placé dans l’axe de la seconde source (figure 3.6). Les réponses impulsionnelles sont calculées sous Matlab avec une méthode de source-image [45] à une fréquence d’échantillonnage de 16 kHz. La réponse impulsionnelle la plus longue calculée possède 5608 coefficients, nombre qui est choisi comme longueur des filtres estimés. Chaque source génère un bruit blanc centré de variance unitaire, décorrélé avec les entrées. Le signal mesuré par le microphone est composé de la somme des contributions des six sources à laquelle est ajouté un bruit blanc centré, indépendant des entrées, tel que le rapport signal sur bruit final soit de 30 dB.

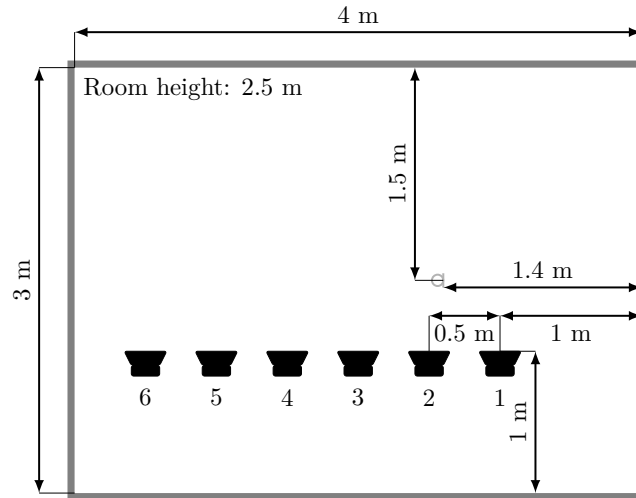


FIGURE 3.6 – Géométrie simulée. Les six haut-parleurs (noir) sont espacés de 50 cm, le microphone apparaît en gris clair.

Deux analyses sont respectivement effectuées, après 5 s et 30 s de mesure. Pour le filtre LMS, nous définissons le pas de convergence comme  $\mu^{(l)} = 1/(NL + 1)$ , correspondant également aux valeurs initiales des vecteurs  $\mu_0^{(l)}$  d'Autostep. Les courbes d'apprentissage sont affichées en figure 3.7. Au départ (jusqu'à 20 s), les deux algorithmes présentent les mêmes performances. Ensuite, nous observons que le filtre LMS converge vers un plateau (dépendant du choix du paramètre  $\mu$ ), alors que l'algorithme Autostep continue de minimiser l'erreur grâce au caractère adaptatif des valeurs de  $\mu^{(l)}$ .

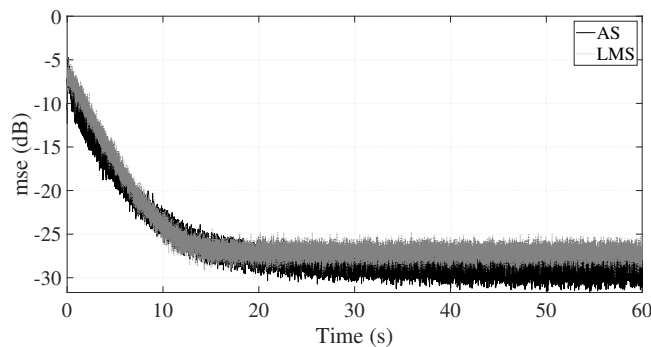
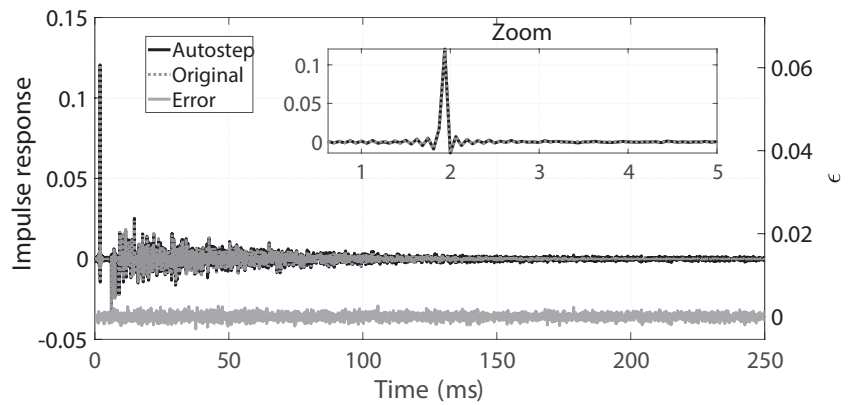


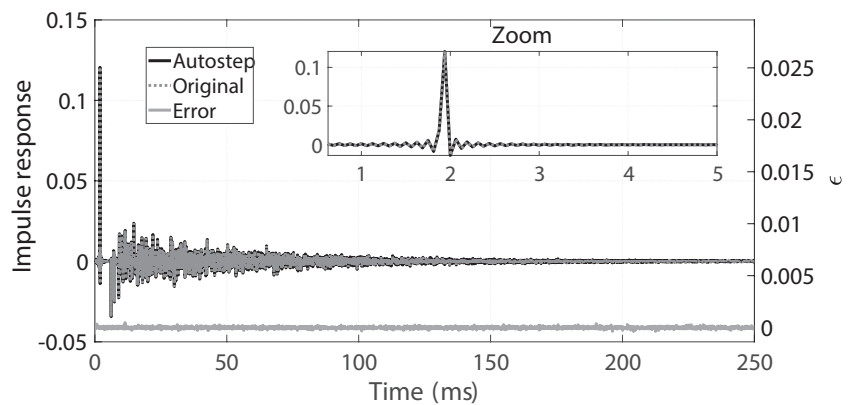
FIGURE 3.7 – *Learning curves* avec  $K = 100$  dans (3.20). Gris : MISO-LMS avec  $\mu^{(l)} = 1/(NL + 1)$ . Noir : MISO-Autostep avec  $\kappa = 0.01$ ,  $\gamma = 10$ .

Il est à noter qu'après 5 s, les deux algorithmes peuvent ne pas avoir fini de converger (figure 3.7) vers leur valeur finale. Cependant, l'estimation peut être suffisante pour des

applications qui ne nécessitent qu'une approximation dans un temps contraint. Ceci est montré en figures 3.8a et 3.9a, respectivement pour les haut-parleurs 1 et 6. Pour visualiser plus précisément les performances, un zoom du début des réponses impulsionnelles est inséré sur chaque figure. Même si les estimations ne sont pas parfaites, les résultats obtenus sont déjà intéressants. Cependant, à cause de la réverbération, la réponse impulsionnelle du haut-parleur le plus éloigné est plus bruitée que celle du haut-parleur le plus proche. Les résultats pour les haut-parleurs 1 et 6 après 60 s sont également donnés sur les figures 3.8b et 3.9b. Comme espéré, après un temps plus long, l'erreur d'estimation est grandement réduite et l'algorithme MISO-Autostep permet d'obtenir une erreur plus faible que celle de MISO-LMS.

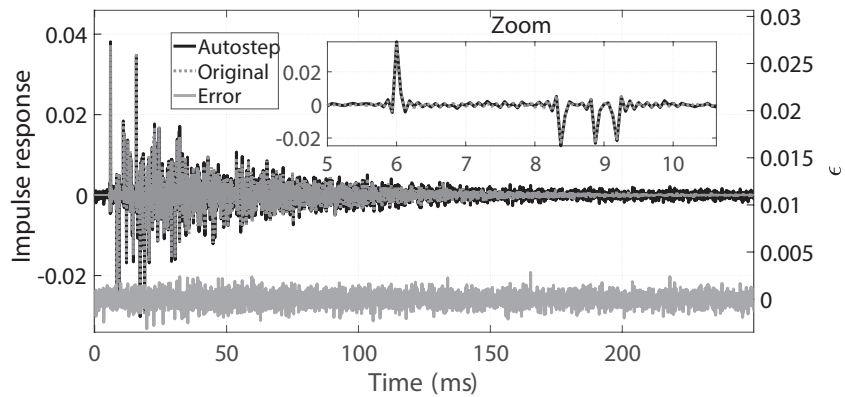


(a) Après cinq secondes de mesure.

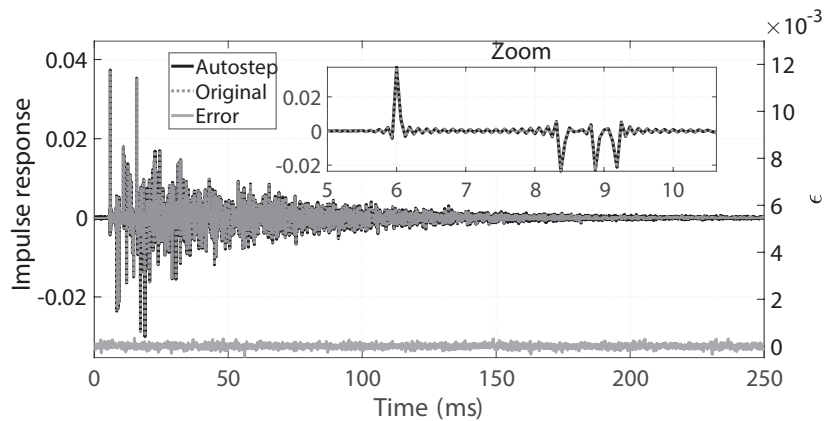


(b) Après soixante secondes de mesure.

FIGURE 3.8 – Réponse impulsionnelle (RI) du haut-parleur 1 dans la salle simulée : (a) après 5 s, (b) après 60 s. La vraie RI est en gris, alors que la RI estimée avec MISO-Autostep est en noir ( $\kappa = 0.01$ ,  $\gamma = 10$ ). L'erreur  $\epsilon$  à la fin de la simulation est en bas des graphiques.



(a) Après 5 s de mesure.



(b) Après 60 s de mesure.

FIGURE 3.9 – Réponse impulsionnelle (RI) du haut-parleur 6 dans la salle simulée : (a) après 5 s, (b) après 60 s. La vraie RI est en gris, alors que la RI estimée avec MISO-Autostep est en noir ( $\kappa = 0.01$ ,  $\gamma = 10$ ). L'erreur  $\epsilon$  à la fin de la simulation est en bas des graphiques.

Les FRF théoriques entre 100 et 2000 Hz des haut-parleurs 1 et 6 sont également calculées (nommées original sur les figures), puis elles sont comparées aux FRF issues des algorithmes Autostep et LMS. Les figures 3.10 et 3.11 comparent les modules des FRF originales et celles d'Autostep des haut-parleurs 1 et 6. On remarque que l'erreur est faible et constante sur toute la bande de fréquence.

Une synthèse des métriques après 60 s est proposée Table 3.2 afin de pouvoir comparer les performances des deux algorithmes. Pour le *misalignemnt*, l'algorithme MISO-Autostep est plus performant de 3,6 dB que MISO-LMS. Concernant l'erreur spectrale, MISO-Autostep est 5 fois plus performant que MISO-LMS.

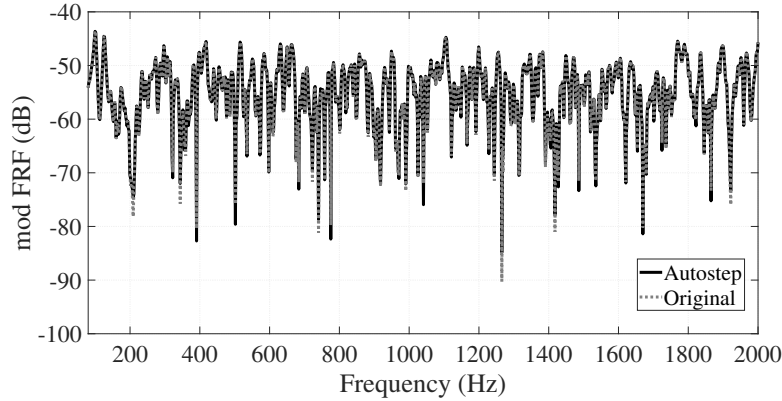


FIGURE 3.10 – FRF (module) du haut-parleur 1 dans la salle simulée. Gris : vraie FRF. Noir : estimée avec Autostep après 60 s.

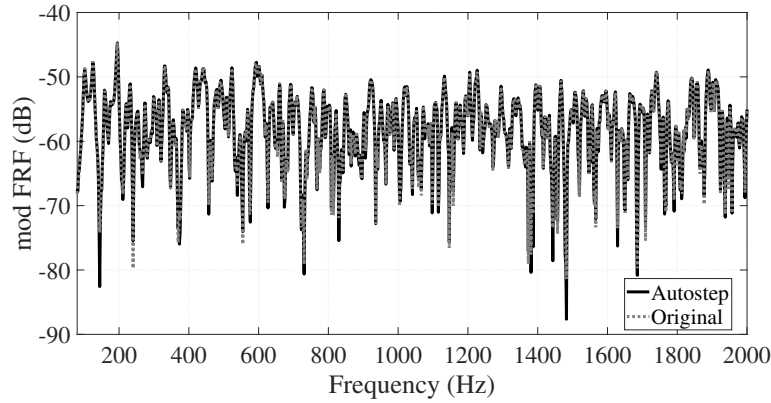


FIGURE 3.11 – FRF (module) du haut-parleur 6 dans la salle simulée. Gris : vraie FRF. Noir : estimée avec Autostep après 60 s.

TABLE 3.2 – Désalignement et erreur spectrale quadratique moyenne dans la salle simulée après 60 s. Moyenne sur tous les haut-parleurs, valeurs min et max.

	Autostep	LMS
$\langle \text{mis} \rangle$ (dB)	-41,4	-37,8
min/max mis (dB)	-41,6/-41,3	-38/-37,7
$\langle \text{mse}_{\text{DFT}} \rangle$ ( $\times 10^{-4}$ )	8,12	42
min/max $\text{mse}_{\text{DFT}}$ ( $\times 10^{-4}$ )	5,68/13	29/64

### 3.4 Validation expérimentale

Cette section présente les résultats de deux campagnes de mesures réalisées dans deux salles : une salle semi-anéchoïque et une salle du laboratoire sans propriété spécifique.

### 3.4.1 Dispositif expérimental

Le dispositif expérimental est contrôlé par une interface audio RME Madiface XT qui permet la synchronisation de l'acquisition et la reproduction. Les sources audio utilisées sont des enceintes à évent 2 voies fabriquées au LAUM. Elles sont alimentées par des signaux issus d'un convertisseur RME DA32 qui sont amplifiés par des modules de puissance quatre voies HPA D604. Le champ de pression résultant est mesuré par un microphone Brüel & Kjær 4958 connecté à son conditionneur puis numérisé par un convertisseur RME AD32. Pour toutes les configurations testées dans cette section, chaque réponse impulsionnelle est au préalable mesurée individuellement à l'aide de la méthode ESS (voir section 2.1). L'acquisition et la reproduction sont effectuées à une fréquence d'échantillonnage de 48 kHz.

### 3.4.2 Mesures en salle semi-anéchoïque

La première mesure est effectuée en salle semi-anéchoïque. Seize haut-parleurs sont disposés régulièrement autour d'un carré de 3 m de côté à 1 m du sol (figure 3.12). Le microphone est placé au centre du carré. Le temps d'acquisition est de 60 s et les réponses impulsionnelles estimées sont composées de 5000 coefficients (100 ms). Cette durée est choisie pour être plus grande que la plus longue des réponses impulsionnelles. Chaque source émet un bruit blanc centré de variance unitaire, indépendant donc décorrélé des autres entrées.

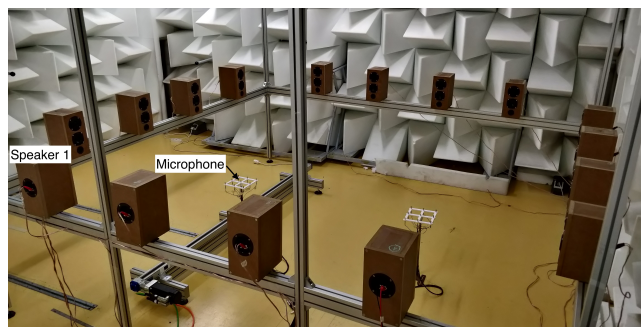


FIGURE 3.12 – Dispositif expérimental installé dans la salle semi-anéchoïque.

La réponse impulsionnelle du haut-parleur 1 (voir position sur la figure 3.12) après 60 s est affichée sur la figure 3.13 et le module de la FRF associée est affichée sur la figure 3.14.

L'estimation par Autostep est très proche de la référence (i.e. ESS) autant pour la réponse impulsionnelle que pour la réponse fréquentielle.

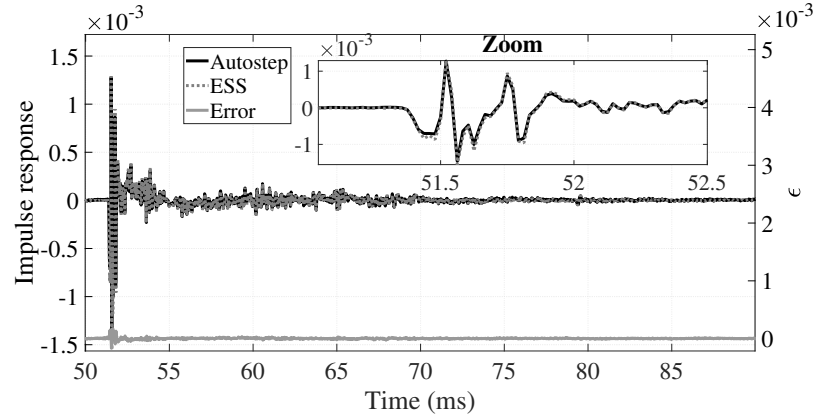


FIGURE 3.13 – Réponse impulsionnelle du haut-parleur 1 dans la salle semi-anéchoïque. Gris : mesurée avec un signal ESS. Noir : estimée avec MISO-AS après 60 s ( $\kappa = 0.01$ ,  $\gamma = 10$ ). Un zoom sur le début de la RI est proposé. L'erreur  $\epsilon$  est également fournie en bas du graphique (axe des ordonnées à droite de la figure).

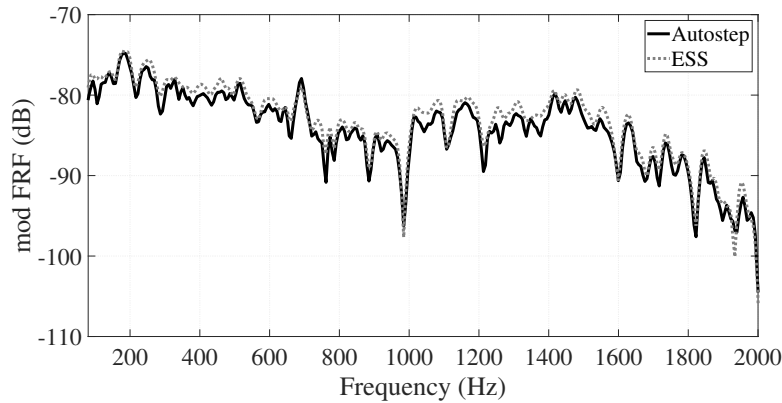


FIGURE 3.14 – FRF (module) du haut-parleur 1 dans la salle semi-anéchoïque. Gris : mesurée avec un signal ESS. Noir : estimée avec MISO-AS après 60 s.

Les métriques après 6 s sont données dans la Table 3.3 (moyenne sur tous les haut-parleurs, valeurs minimale et maximale) afin de pouvoir comparer les performances des deux algorithmes. Concernant le *misalignment*, Autostep offre une performance 2,6 dB inférieure à celle de LMS. Dans le domaine fréquentiel, Autostep permet d'obtenir une erreur réduite d'un facteur 1.4 et une dispersion réduite d'un facteur 2.7 par rapport à LMS.



TABLE 3.3 – Désalignement et erreur spectrale quadratique moyenne dans la salle semi-anéchoïque après 6 s. Moyenne sur tous les haut-parleurs, valeurs min et max.

	Autostep	LMS
$\langle \text{mis} \rangle$ (dB)	-63,3	-60,7
min/max mis (dB)	-66,3/-61,7	-62,0/-59,8
$\langle \text{mse}_{\text{DFT}} \rangle$	0,053	0,075
min/max $\text{mse}_{\text{DFT}}$	0,045/0,066	0,054/0,11

### 3.4.3 Mesures dans une salle du laboratoire

La seconde série de mesures est effectuée dans une salle du laboratoire où la durée de réverbération  $RT_{30}$  (autour de 0,8 s) est relativement constante sur la plage [100 Hz - 5 kHz]. Huit haut-parleurs sont alignés sur une paillasse. Un microphone BK 1/2" 4192 est positionné au milieu de la pièce, dans l'axe du premier haut-parleur (figure 3.15).

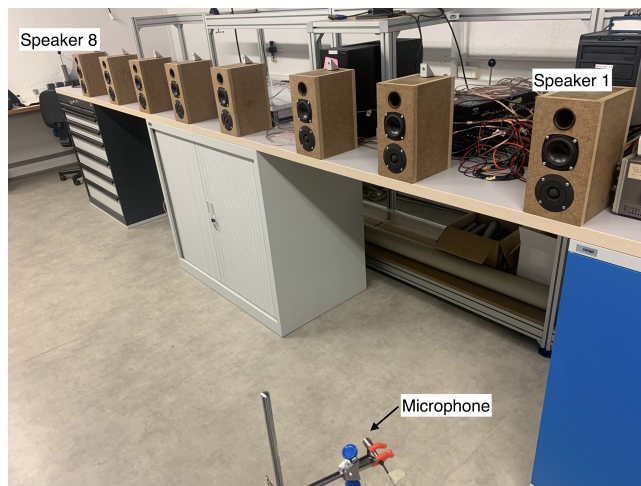


FIGURE 3.15 – Setup expérimental dans la salle du laboratoire.

Les réponses impulsionnelles après 6 min du haut-parleur 1 (le plus proche du microphone) et du haut-parleur 8 (le plus éloigné du microphone) sont tracées sur les figures 3.16 and 3.17 respectivement. La mesure du haut-parleur 1, situé à proximité du microphone, présente une réponse impulsionnelle très piquée, illustrant le fait que le champ direct est prépondérant par rapport au champ réverbéré. Le temps de convergence est toutefois plus long qu'en simulation à cause de la complexité plus importante de l'environnement. La réponse impulsionnelle du haut-parleur 8 présente un temps de réverbération plus long, et l'erreur est plus grande que celle de la réponse impulsionnelle 1.

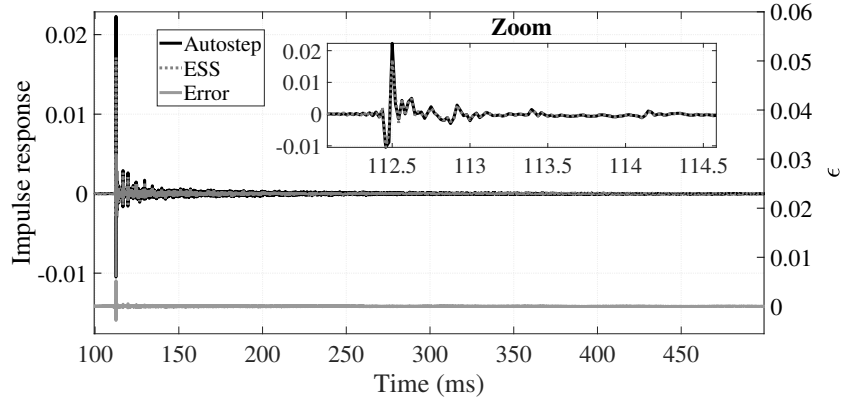


FIGURE 3.16 – Réponse impulsionnelle du haut-parleur 1 dans la salle du laboratoire. Gris : mesurée avec un signal ESS. Noir : estimée avec MISO-AS après 60 s ( $\kappa = 0.01$ ,  $\gamma = 10$ ). Un zoom sur le début de la RI est proposé. L’erreur  $\epsilon$  est également fournie en bas du graphique (axe des ordonnées à droite de la figure).

Les modules des FRF des haut-parleurs 1 et 8 sont donnés sur les figures 3.18 et 3.19 respectivement. Une fois encore, la méthode Autostep donne une estimation très proche de la référence (ESS). Les métriques après 6 min sont données Table 3.4. Concernant le *misalignment*, la méthode Autostep a une erreur de 2,2 dB plus faible que LMS. Dans le domaine fréquentiel, l’erreur de la méthode Autostep est 1.5 fois plus petite et la dispersion est quant à elle 2.3 fois plus petite.

TABLE 3.4 – Désalignement et erreur spectrale quadratique moyenne dans la salle du laboratoire après 6 min. Moyenne sur tous les haut-parleurs, valeurs min et max.

	Autostep	LMS
$\langle \text{mis} \rangle$ (dB)	-47,6	-45,4
min/max mis (dB)	-54,4/-41,9	-48,7/-41,2
$\langle \text{mse}_{\text{DFT}} \rangle$	0,071	0,11
min/max $\text{mse}_{\text{DFT}}$	0,058/0,08	0,09/0,14

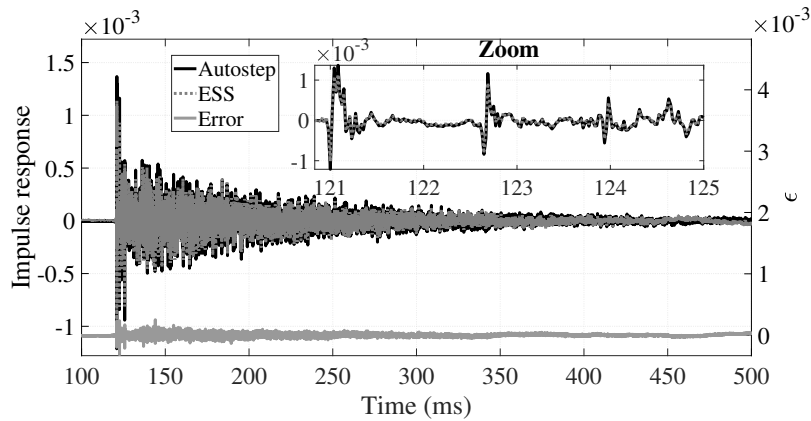


FIGURE 3.17 – Réponse impulsionnelle du haut-parleur 8 dans la salle du laboratoire. Gris : mesurée avec un signal ESS. Noir : estimée avec MISO-AS après 60 s ( $\kappa = 0.01$ ,  $\gamma = 10$ ). Un zoom sur le début de la RI est proposé. L'erreur  $\epsilon$  est également fournie en bas du graphique (axe des ordonnées à droite de la figure).

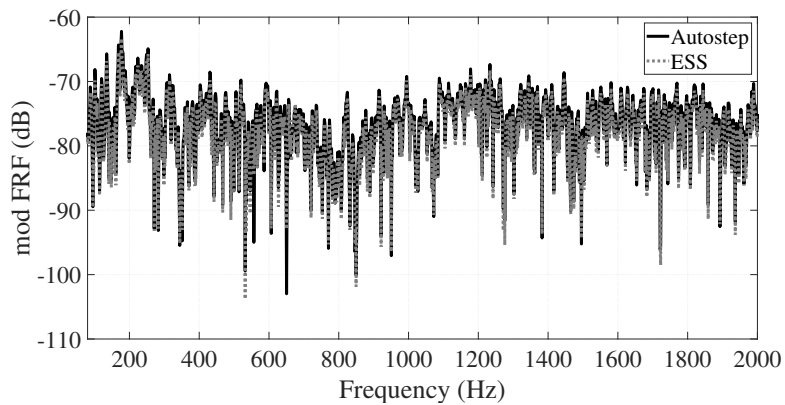


FIGURE 3.18 – FRF (module) du haut-parleur 1 dans la salle du laboratoire. Gris : mesurée avec un signal ESS. Noir : estimée avec MISO-AS après 6 min.

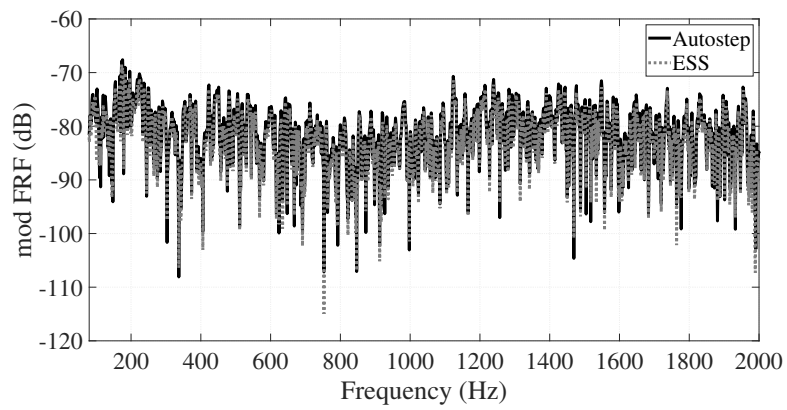


FIGURE 3.19 – FRF (module) du haut-parleur 8 dans la salle du laboratoire. Gris : mesurée avec un signal ESS. Noir : estimée avec MISO-AS après 6 min.

## 3.5 Conclusion

L'algorithme MISO-AS obtient des performances égales voire supérieures à l'algorithme MISO-LMS. Cependant, sa faible dépendance aux méta-paramètres permet d'utiliser cet algorithme sur des systèmes MISO où les RI peuvent être très différentes entre elles.

Il est à noter que dans un cas réel, où le bruit de mesure est plus important qu'en simulation, les temps de convergences sont alors bien plus importants. C'est le cas de la mesure dans la salle réelle (la mesure ayant été effectuée un jour de pluie, les hypothèses de bruit blanc se sont avérées non valides), ce qui explique en partie un temps de convergence de 6 min.

Enfin, il convient de souligner que la faible dépendance de l'algorithme MISO-Autostep à l'égard de ses méta-paramètres ouvre de nouvelles perspectives en matière d'identification des systèmes, en particulier la caractérisation d'environnements non stationnaires, y compris les systèmes linéaires MISO variant dans le temps.

# EXTENSION AUX SYSTÈMES VARIANT DANS LE TEMPS

---

Dans ce chapitre, l'algorithme MISO-Autostep est testé dans une configuration variant dans le temps en guise de perspective à cette première partie. Au préalable, quelques rappels et explications sur la méthode de simulation des systèmes variant dans le temps sont présentés.

## 4.1 Systèmes linéaires variant dans le temps (LTV)

Dans cette section est présentée la méthode de simulation d'un système variant dans le temps, avec la présentation de la convolution associée.

Supposons maintenant que la réponse impulsionnelle change au cours du temps et définissons alors les instants suivants<sup>1</sup> :

- $n$  représente l'instant courant d'observation connu jusqu'à présent ;
- $m$  représente l'instant d'entrée du signal dans le filtre (le filtre changeant au cours du temps, il est logique que la réponse impulsionnelle change avec l'instant d'entrée dans le filtre).

La convolution entrée-sortie s'écrit alors :

$$p(n) \stackrel{\text{def}}{=} \sum_{m=-\infty}^{\infty} h(m, n)x(m). \quad (4.1)$$

Il convient par suite d'analyser la causalité dans ce contexte :

- la causalité « simple » impose une réponse impulsionnelle nulle pour tout instant d'observation  $n$  négatif :  $\forall n < 0, h(m, n) = 0$  ;

---

1. Les notations des temps ( $k, l, m$ ) sont celles de la littérature, et sont propres à ce chapitre 4.

- la réponse impulsionnelle prise à un instant d'observation  $n$  doit être nulle pour tous les instants correspondant à des instants futurs d'entrée du signal  $m$ , soit  $\forall n < m, h(m, n) = 0$ .

En posant alors  $l = n - m$ , nous avons donc :

$$p(n) = \sum_{l=0}^{N-1} h(n-l, n)x(n-l), \quad (4.2)$$

convolution qui sera notée par la suite :

$$p(n) = (h_k * x)(n), \quad (4.3)$$

l'indice  $k$  dénotant une dépendance au temps de la réponse impulsionnelle.

De la même façon que pour les systèmes linéaires invariant dans le temps, cette convolution peut se mettre sous forme matricielle :

$$\begin{pmatrix} p(0) \\ p(1) \\ p(2) \\ \vdots \\ p(N-1) \end{pmatrix} = \begin{pmatrix} h(0,0) & 0 & \cdots & 0 \\ h(0,1) & h(1,1) & \ddots & 0 \\ \vdots & h(1,2) & \ddots & \vdots \\ \vdots & \ddots & \ddots & \vdots \\ h(0,N-1) & \cdots & h(N-2,N-1) & h(N-1,N-1) \end{pmatrix} \begin{pmatrix} x(0) \\ x(1) \\ x(2) \\ \vdots \\ x(N-1) \end{pmatrix}. \quad (4.4)$$

La matrice n'est donc plus une matrice de TOEPLITZ, sauf si toutes les réponses impulsionnelles sont invariantes dans le temps :  $\forall m \in \mathbb{N}, h(m, n) = h(n)$ .

Il y a cependant des précautions à prendre liées à cette approche :

- la convolution ainsi définie n'est pas commutative :  $h_1(m, n) * h_2(m, n) \neq h_2(m, n) * h_1(m, n)$  ;
- elle reste cependant associative et distributive.

Nous utiliserons cette technique de convolution pour les systèmes variant dans le temps par la suite.

## 4.2 Test de l'algorithme MISO-AS dans un cas LTV

La simulation est un changement brusque dans la géométrie du système. Ainsi, à l'instant du changement, les réponses impulsionnelles estimées sont fausses et l'algorithme doit partir de conditions complètement fausses pour estimer les nouvelles réponses impulsionnelles.

Nous reprenons la géométrie de la section 3.3 avec le microphone placé au départ devant le haut-parleur 2. La simulation complète dure 1 min, et à 30 s le microphone est déplacé pour se situer devant le haut-parleur 5. Les autres paramètres restent inchangés (RT60 de 0,5 s, fréquence d'échantillonnage de 16 kHz,  $N = 5608$ , RSB de 30 dB). Les *learning curves* sont obtenues par la méthode de Monte Carlo avec 100 essais, puis lissées par une moyenne mobile sur 100 points.

Les courbes de convergence de l'algorithme MISO-Autostep et MISO-LMS sont présentées sur la figure 4.1. Sur la première moitié de l'expérience, nous retrouvons la même courbe que sur la figure 3.7 pour l'algorithme MISO-Autostep, ce qui est normal puisque c'est la même simulation. À partir de 30 s l'erreur augmente brusquement, autant pour MISO-Autostep que MISO-LMS. Après 20 s les algorithmes MISO-Autostep et MISO-LMS ont fini de converger vers les solutions correspondant à la nouvelle configuration.

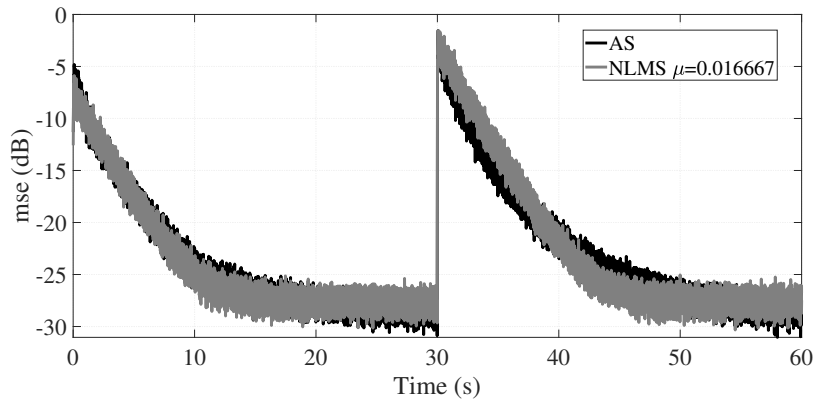


FIGURE 4.1 – *Learning curves* avec  $K = 100$  dans (3.20) pour un changement brusque. Gris : MISO-LMS avec  $\mu^{(l)} = 1/(NL + 1)$ . Noir : MISO-Autostep avec  $\kappa = 0.01$ ,  $\gamma = 10$ .



# CONCLUSION

---

Comme précisé en introduction, les algorithmes de *sound zones* actuels nécessitent *a priori* la connaissance des réponses impulsionnelles (ou FRF pour leurs versions fréquentielles). Dans le cas variant dans le temps, les réponses impulsionnelles doivent également être réajustées au cours du temps pour s'adapter aux changements.

L'algorithme MISO-Autostep permet d'estimer les réponses impulsionnelles (chapitre 3) sans avoir à régler finement les paramètres de l'algorithme. De plus, dans le cas où le système varie au cours du temps (température, changements dans la pièce, déplacement des zones), il est capable de mettre à jour les réponses impulsionnelles au cours du temps (chapitre 4).

Les temps de convergence peuvent être longs, surtout si le rapport signal sur bruit est faible. Cependant, des travaux récents sur les *sound zones* ont montré qu'une estimation même grossière des réponses impulsionnelles permet de générer des zones avec des performances intéressantes [14].

Dans la partie suivante, des algorithmes de *sound zones* seront présentés, et en dernière partie de ce manuscrit, des premiers résultats de *sound zones* mobiles seront analysés, utilisant l'algorithme MISO-Autostep développé dans cette première partie.

DEUXIÈME PARTIE

# Sound zones

---

# CONTEXTE & ÉTAT DE L'ART

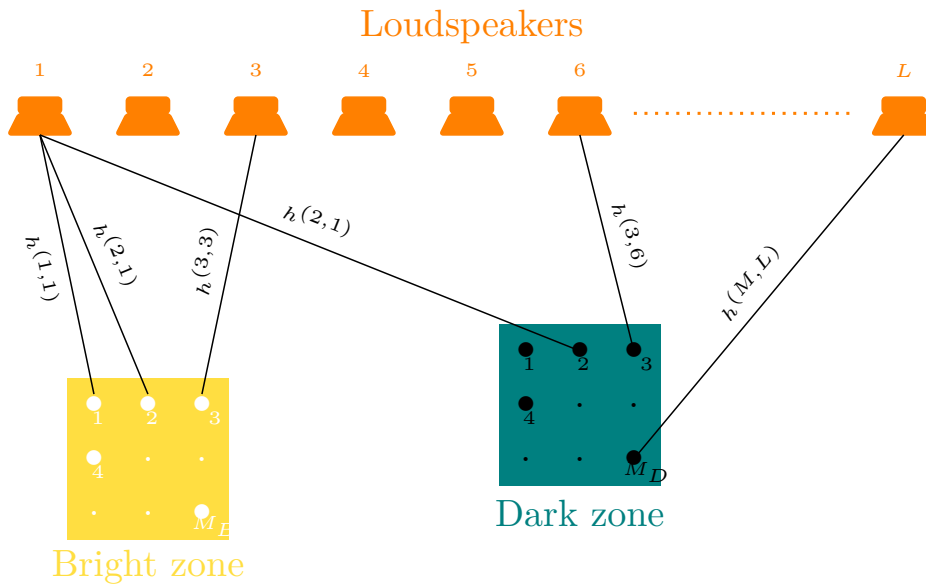
---

## 6.1 Introduction

L'objectif d'un système de *sound zones* est de contrôler le champ acoustique dans différentes zones de l'espace. Le système le plus simple de *sound zones* consiste à définir deux zones : une zone où l'on souhaite diffuser un contenu audio, dite *bright zone*, et une zone idéalement silencieuse nommée *dark zone* [7]. Par superposition, pour diffuser deux contenus audio différents dans chaque zone, il suffit de résoudre à nouveau le problème en intervertissant les deux zones précédentes. Parmi les applications existantes, nous pouvons citer des utilisations dans l'informatique [46], la téléphonie [11], l'industrie automobile [9], la télévision [47] ou encore l'aéronautique [10].

Il existe de nombreuses technologies qui permettent d'isoler des zones de l'espace de nuisances sonores ou cibler des zones à sonoriser [8]. On peut citer entre autres les antennes paramétriques [48], les sources directives [49], les matériaux isolants [50]. Cependant, ces méthodes souffrent des performances médiocres pour les grandes longueurs d'onde, i.e. à basse fréquence. Des méthodes de reconstruction de champs acoustiques existent également, basées sur des équations intégrales aux frontières analytiques dérivées de la théorie de KIRCHHOFF-HELMHOLTZ, comme la *Wave Field Synthesis* (WFS) [51]-[53] ou l'ambisonie d'ordres supérieurs (*High Order Ambisonie* - HOA) [54], [55]. Ces méthodes nécessitent cependant de nombreuses sources et sont sensibles à leur placement. Les méthodes de *sound zones* sont quant à elles basées sur un problème d'optimisation sous contraintes.

La figure 6.1 présente un schéma classique pour la génération de *sound zones*. Pour reproduire le (ou les) contenu(s) audio(s), un réseau de  $L$  haut-parleurs est utilisé (voir l'annexe B pour le choix de la géométrie). Chaque zone est discrétisée par une antenne de  $M_B$  et  $M_D$  microphones, respectivement pour la *bright zone* et la *dark zone*.

FIGURE 6.1 – Schéma de principe pour les *sound zones*.

Notons que les champs acoustiques sont contrôlés uniquement dans les zones d'intérêt.

Connaissant les chemins acoustiques entre chaque haut-parleur et chaque microphone, représentés par les réponses impulsionnelles dans le domaine temporel ou les réponses en fréquence dans le domaine de Fourier, l'algorithme de *sound zones* doit trouver  $L$  filtres à appliquer aux  $L$  haut-parleurs pour obtenir les  $L$  signaux à émettre. Une fois que ces  $L$  signaux sont émis, ils interfèrent entre eux et génèrent ainsi les différentes zones.

Comme précisé dans le paragraphe précédent, il existe deux domaines pour représenter le comportement du système : le domaine temporel ou le domaine fréquentiel. Une partie importante de la littérature présente des algorithmes de *sound zones* dans le domaine fréquentiel [7], [56]-[58], où le coût de calcul est très souvent bien plus faible que dans le domaine temporel. De plus, le calcul fréquence par fréquence permet un réglage fin de l'algorithme puisqu'il est possible de régulariser (si le problème est mal conditionné) chaque fréquence individuellement. Pour autant, des problèmes de causalité des filtres peuvent apparaître quand le problème est résolu dans le domaine fréquentiel, ce qui peut simplement être évité par les méthodes agissant dans le domaine temporel [59].

Une fois le système modélisé, que ce soit dans le domaine temporel ou fréquentiel, le problème d’optimisation doit être posé. Il est alors possible de classer les algorithmes de la littérature dédiés à cette étape d’optimisation en deux grandes familles :

- La première famille tente de maximiser le rapport des énergies potentielles acoustiques dans les zones, et propose les algorithmes dits *Acoustic Contrast Control* [7], notés par la suite ACC.
- La seconde famille est construite autour des algorithmes de *Pressure Matching* [60], en référence à une pression cible désirée dans la *bright zone* (l’amplitude et la phase sont contraintes) et notés usuellement PM. Cette famille résout un problème de moindres carrés dans la *bright zone* en complétant le problème d’optimisation par des contraintes dans la *dark zone*<sup>1</sup>.

Une classe d’algorithmes plus récents, *Amplitude Matching* [64], propose de minimiser l’erreur de reproduction en amplitude seulement (sans la phase) dans la *bright zone* tout en minimisant l’énergie de la *dark zone*. Cependant, ces algorithmes ne contraignent pas la phase dans la *bright zone* et restent proches des algorithmes ACC d’un point de vue perceptif [58]. D’autres algorithmes partent des algorithmes ACC et tentent d’en corriger les défauts. Une publication récente propose ainsi de contraindre l’intensité acoustique dans la *bright zone* tout en maximisant le niveau relatif entre les zones, i.e. le contraste [65]. En effet, l’algorithme ACC maximise le contraste, sans fixer un niveau cible dans la *bright zone*. De plus, l’algorithme ACC s’appuie sur l’énergie potentielle acoustique moyenne<sup>2</sup> sur la zone, alors que la distribution spatiale de l’énergie dans la zone peut être très éloignée d’une distribution uniforme. Pour limiter ces effets, des contraintes de planarité de la zone peuvent être ajoutées [58].

Le champ acoustique généré par les haut-parleurs et mesuré par les microphones peut être réécrit sur une base modale (celle des harmoniques sphériques par exemple), puis les problèmes d’optimisation (type ACC ou PM) peuvent être résolus sur cette nouvelle base [66]-[69].

---

1. Les problèmes d’optimisation liés aux *sound zones* s’inscrivent dans une théorie plus large de traitement de la parole *Variable Span Linear Filtering* (VSLF) [61], [62]. En 2018, LEE *et al.* [63] ont montré que les deux familles d’algorithmes de *sound zones*, *Acoustic Contrast Control* (ACC) et *Pressure Matching* (PM), sont deux cas extrêmes des algorithmes VSLF. Les algorithmes qui s’inspirent des deux cas précédents, comme la solution combinée de SIMON-GALVEZ [59], peuvent également être vus comme un cas particulier d’algorithme VSLF.

2. Approximée ici par  $e_p = \frac{1}{KM_B} \sum_{k=0}^{K-1} \sum_{m=1}^M p^{(m)}(k)p^{(m)}(k)$

Depuis quelques années, des études psychoacoustiques sont conduites pour valider les qualités perceptives des différents algorithmes présents dans la littérature. En 2014, une première étude [70] s'est intéressée à la qualité de reproduction du contenu audio dans la *bright zone*. En effet, les algorithmes ACC, en maximisant le rapport des énergies potentielles dans les zones, ne contraignent pas la phase dans la *bright zone* au contraire des algorithmes PM. Il ressort de cette étude que les algorithmes PM sont perceptivement plus appréciés que les algorithmes ACC. En 2017, RAMO [71] a quant à lui étudié la performance des algorithmes du point de vue perceptif. Dans la littérature, un indicateur de performance très largement utilisé est le contraste entre les zones, celui-ci étant défini comme le rapport entre l'énergie du signal désiré dans la zone d'écoute et l'énergie due à ce signal dans l'autre zone. Un « bon » algorithme de *sound zones* doit permettre à un utilisateur d'écouter son signal audio sans être gêné par d'autres signaux que pourraient écouter d'autres utilisateurs. Cette gêne est nommée par RAMO distractibilité<sup>3</sup>, et il ressort de l'étude qu'un contraste d'au moins 30 dB est nécessaire pour s'assurer d'une bonne isolation entre les zones. Une publication de 2018 propose d'ailleurs un algorithme de *sound zones* basé sur un critère perceptif pour générer les zones [74]. Ici, c'est une contrainte de masquage entre les signaux qui est utilisée. Cet algorithme a été modifié récemment en prenant en compte le signal d'entrée et en créant des filtres de *sound zones* variant dans le temps par DE KOEIJER *et al.* [75].

## 6.2 Etat de l'art des méthodes fonctionnant dans le domaine temporel

Pour la même raison que celle évoquée dans la première partie (estimation de réponses impulsionnelles de systèmes MISO, cf. section 2.3), nous avons choisi de nous restreindre aux algorithmes fonctionnant dans le domaine temporel, en particulier pour leur faculté à pouvoir être étendus au cas des systèmes variant dans le temps. Il est à noter que la solution développée dans le domaine temporel peut différer de la solution trouvée dans le domaine fréquentiel, notamment à cause du paramètre de régularisation [76].

---

3. La distractibilité est la capacité d'un programme audio à attirer l'attention par rapport à un autre programme [72]. Dans les *sound zones*, le programme résiduel dans la *bright zone* non désiré doit être masqué par le programme principal. Cependant, en fonction des types de contenu audio, les contrastes doivent être différents (un signal de parole ayant tendance à être plus difficile à masquer qu'un programme musical) [73].

Les premiers travaux de contrôle d’un champ acoustique dans le domaine temporel datent des années 1990, notamment avec les travaux pionniers de KIRKEBY [2], [77]. À cette époque, une seule zone de l’espace est contrôlée et l’algorithme présenté minimise l’erreur de reproduction dans la zone d’intérêt.

Les deux familles d’algorithmes de *sound zones* développées dans le domaine fréquentiel ont ensuite été adaptées au domaine temporel :

- Broad-band Acoustic Contrast Control (BACC) [78] ;
- Broad-band Acoustic Contrast Control with Pressure Matching (BACC-PM) [59].

L’algorithme BACC résout un problème aux valeurs propres [59], [78]-[80]. L’objectif est de maximiser le rapport entre l’énergie potentielle dans la *bright zone* et l’énergie potentielle dans la *dark zone*. L’algorithme BACC-PM est quant à lui un problème de moindres carrés, où l’objectif est d’atteindre une pression cible dans la *bright zone* tout en ajoutant une contrainte de pression minimale dans la *dark zone*. L’algorithme BACC présente les meilleures performances en termes de contraste entre les zones. Par contre, l’algorithme BACC-PM a un meilleur rendu dans la *bright zone* en contrôlant la phase dans cette zone. Pour concilier ces deux critères, une solution combinée a été proposée par GALVÈZ *et al.* [59], en intégrant un paramètre  $\beta$  qui permet de passer de BACC-PM ( $\beta = 0$ ) à BACC ( $\beta = 1$ ), avec toutes les variations entre les deux valeurs extrêmes. Les algorithmes BACC et BACC-PM seront présentés plus en détail dans les sections 6.5 et 6.6 respectivement.

Le principal problème de la formulation dans le domaine temporel des *sound zones* est la taille des matrices qui peut devenir énorme par rapport à une formulation fréquentielle. Pour l’algorithme BACC-PM, la matrice à inverser est ainsi carrée, de taille  $LN \times LN$ , avec  $N$  l’ordre des filtres et  $L$  le nombre de haut-parleurs [59], à comparer avec la taille  $L \times L$  pour sa version fréquentielle [60]. Or l’inversion d’une matrice carrée de taille  $N$  a un coût de calcul en  $O(N^3/3)$ . Pour pallier cette difficulté, de nouvelles méthodes sont apparues dans la littérature ces dernières années. Une première approche est de limiter la taille de la matrice à inverser en limitant l’ordre des filtres et/ou des réponses impulsionnelles [81], [82]. Selon la théorie *Variable span trade-off filter* (VAST), les algorithmes de *sound zones* peuvent être déduits d’un problème aux valeurs propres généralisés, qu’ils soient de type PM ou ACC. Des techniques de modélisation dans un espace de rang plus petit (*low-rank modeling*) peuvent donc être utilisées pour résoudre ce problème dans un espace de plus

petite dimension [83]. Enfin, la technique de résolution du problème d'optimisation peut être modifiée pour s'affranchir de l'inversion d'une matrice, par exemple en utilisant une descente de gradient conjugué [84].

Une publication récente [85] propose une méthode de génération de *sound zones* différente, en ce sens que ce ne sont plus les conditions sur les pressions ou énergies dans les zones qui pilotent l'algorithme, mais c'est le rapport signal-interférence<sup>4</sup> qui est optimisé.

Enfin, des études ont été menées sur l'influence du bruit de mesure des réponses impulsionnelles sur les métriques des *sound zones* [86], [87].

Par la suite, nous nous restreindrons dans ce document à l'algorithme BACC-PM, et nous tenterons d'obtenir un contraste cible d'au moins 30 dB.

### 6.3 Sound zones dynamiques

À ce jour, la plupart des méthodes de *sound zones* sont destinées à des utilisateurs statiques. Or, pouvoir utiliser ces méthodes dans un contexte dynamique, par exemple en s'adaptant à la position de l'auditeur, permettrait d'ouvrir le champ des applications à un plus large éventail de contextes. En 2019, ROUSSEL [88] développe plusieurs algorithmes dans le domaine de FOURIER dans l'optique de rendre les *sound zones* mobiles mais, à nouveau, les contraintes de temps réel ne permettent pas d'atteindre les objectifs attendus.

Plus récemment, VINDROLA *et al.* [16] proposent une méthode adaptative afin de réajuster les filtres de *sound zones* dans le cas d'un déplacement de l'utilisateur au sein de l'habitacle d'une voiture. Partant de positions définies et mesurées, l'algorithme permet alors d'adapter les filtres dans un cas de petits déplacements (quelques centimètres) de l'auditeur par rapport à sa position assise précédente, dans un contexte de véhicule terrestre. La nouveauté de cet algorithme est de s'appuyer sur l'erreur de reproduction mesurée, ce qui permet de le qualifier d'algorithme en boucle fermée, au contraire des algorithmes précédemment proposés dans la littérature, qui, eux, fonctionnent en boucle ouverte. Il est à noter que certains algorithmes sont partiellement corrigés en *feed-forward* pour corriger des perturbations extérieures, comme la température [89].

---

4. Le rapport signal-interférence plus bruit (*SINR*) est le rapport des puissances du signal désiré et du signal non désiré dans chaque zone.



De même, MØLLER *et al.* [15] ont proposé une méthode de résolution de *sound zones* dans une salle, dans le cas d’un système variant dans le temps. Pour cela, ils adaptent le problème d’optimisation (proche d’un BACC-PM) pour le résoudre avec la méthode *moving horizon (MH)*. En considérant ce qu’il s’est passé et en prédisant ce qu’il va se passer, tout en incluant le signal d’entrée, cette technique permet de s’adapter aux caractéristiques du signal à émettre et à d’éventuels changements dans les réponses impulsionnelles. Cela nécessite cependant que ces dernières soient mesurées régulièrement (tous les 10 cm dans leur cas) et la résolution nécessite une inversion matricielle de grande dimension. Pour diminuer le coût de calcul de son algorithme, MØLLER *et al.* [90] proposent une méthode dans le domaine de Fourier, adaptative, pour calculer les filtres des *sound zones* dans le cas variant dans le temps. Notons cependant que les réponses impulsionnelles sont *a priori* connues à chaque instant.

Enfin, récemment, HU *et al.* [14] ont proposé un algorithme de *sound zones* qui estime les réponses impulsionnelles et réactualise les filtres conjointement, au cours du temps. Pour la partie estimation des réponses impulsionnelles, les auteurs utilisent un filtre NLMS, et ils proposent un nouvel algorithme BACC récursif pour résoudre le problème de *sound zones* au cours du temps. La réactualisation des réponses impulsionnelles dans les systèmes d’ANC est un sujet de recherche d’actualité, et un système d’ANC peut être vu comme un système de *sound zones* qui ne contiendrait qu’une seule *bright zone*. HU *et al.* utilisent cette idée en couplant la partie estimation des réponses impulsionnelles d’un système d’ANC [19] à un algorithme de *sound zones*. L’algorithme présenté ajoute cependant deux paramètres, pour les descentes de gradient, qui peuvent rendre instable le système et minorer les performances.

## 6.4 Formulation mathématique du problème

Dans cette section, les équations permettant de calculer les pressions dans chaque zone au cours du temps seront présentées. Selon que l’on considère le signal d’entrée à spectre plat ou non, les équations peuvent être différentes, comme détaillé ci-dessous.

Le système audio utilisé pour la création des *sound zones* est constitué de  $L$  sources réparties autour de la zone à contrôler. La *bright zone* est discrétisée par  $M_B$  microphones et la *dark zone* par  $M_D$  microphones (et nous noterons  $M = M_B + M_D$ ).

La réponse impulsionnelle entre le haut-parleur  $l$  et le microphone  $m$  est notée  $h^{(m,l)}$  et est composée de  $N$  coefficients. Le filtre de *sound zones* appliqué au haut-parleur  $l$  est un filtre FIR d'ordre  $N$  noté  $w^{(l)}$ . Tous les signaux et réponses impulsionnelles sont à valeurs réelles.

La formulation mathématique du système audio se met sous la forme :

$$p^{(m)}(n) = \sum_{l=1}^L \left( h^{(m,l)} * w^{(l)} * x \right) (n), \quad (6.1)$$

avec  $h^{(m,l)}$  la réponse impulsionnelle entre le haut-parleur  $l$  et le microphone  $m$ ,  $w^{(l)}$  le filtre à appliquer au haut-parleur  $l$  et  $p^{(m)}(n)$  la pression au microphone  $m$ ,  $x$  étant le signal à diffuser. Ici, nous considérons connus le signal à diffuser et les réponses impulsionnelles, tout l'enjeu étant de déterminer les filtres  $w^{(l)}$  tels que la pression aux points de mesure respecte le cahier des charges.

Nous noterons par la suite en indice la restriction aux microphones de la *bright zone* par  $(\cdot)_B$  et à ceux de la *dark zone* par  $(\cdot)_D$ .

### 6.4.1 Cas 1 : Sans prendre en compte le signal d'entrée

De nombreux algorithmes sont présentés en considérant le signal d'entrée comme ayant *a priori* un spectre plat, i.e. assimilable à un dirac  $\delta(n)$ .

Nous disposons des RI sous forme de filtres FIR (*Finite Impulse Response*). Ainsi, la réponse impulsionnelle  $h_B^{(m,l)}(n)$  peut être présentée sous la forme d'une matrice de Toeplitz, notée  $\mathbf{H}_B^{(m,l)} \in \mathbb{R}^{2N-1 \times N}$ , pour la convolution :

$$\mathbf{H}_B^{(m,l)} = \begin{bmatrix} h_B^{(m,l)}(0) & 0 & 0 \\ \vdots & \ddots & 0 \\ h_B^{(m,l)}(N-1) & \ddots & h_B^{(m,l)}(0) \\ 0 & \ddots & \vdots \\ 0 & 0 & h_B^{(m,l)}(N-1) \end{bmatrix}_{2N-1 \times N}. \quad (6.2)$$

Définissons la matrice par bloc de réponses impulsionnelles  $\mathbf{H}_B \in \mathbb{R}^{M_B(2N-1) \times LN}$ , contenant tous les couples  $(m, l)$  de matrices  $\mathbf{H}_B^{(m,l)}$  de la *bright zone* :

$$\mathbf{H}_B = \begin{bmatrix} \mathbf{H}_B^{(1,1)} & \cdots & \mathbf{H}_B^{(1,L)} \\ \vdots & \mathbf{H}_B^{(m,l)} & \vdots \\ \mathbf{H}_B^{(M_B,1)} & \cdots & \mathbf{H}_B^{(M_B,L)} \end{bmatrix}_{M_B(2N-1) \times LN}. \quad (6.3)$$

La même démarche pour les réponses impulsionnelles dans la *dark zone* permet de noter la matrice par bloc de réponses impulsionnelles  $\mathbf{H}_D \in \mathbb{R}^{M_D(2N-1) \times LN}$  contenant tous les couples  $(m, l)$  de matrices  $\mathbf{H}_D^{(m,l)}$  de la *dark zone* :

$$\mathbf{H}_D = \begin{bmatrix} \mathbf{H}_D^{(1,1)} & \cdots & \mathbf{H}_D^{(1,L)} \\ \vdots & \mathbf{H}_D^{(m,l)} & \vdots \\ \mathbf{H}_D^{(M_D,1)} & \cdots & \mathbf{H}_D^{(M_D,L)} \end{bmatrix}_{M_D(2N-1) \times LN}. \quad (6.4)$$

Le filtre de *sound zones*  $w^{(l)}(n)$  étant un FIR de  $N$  points, il est possible de regrouper ses coefficients dans le vecteur  $\mathbf{w}^{(l)} \in \mathbb{R}^{N \times 1}$  :

$$\mathbf{w}^{(l)} = \left[ w^{(l)}(0) \quad \cdots \quad w^{(l)}(N-1) \right]_{N \times 1}^\top, \quad (6.5)$$

et par extension le vecteur filtrage  $\mathbf{w} \in \mathbb{R}^{NL \times 1}$  contenant tous les filtres s'écrit :

$$\mathbf{w} = \left[ \mathbf{w}^{(1)\top} \quad \cdots \quad \mathbf{w}^{(L)\top} \right]_{NL \times 1}^\top. \quad (6.6)$$

En assimilant le signal d'entrée  $x(n)$  à une impulsion  $\delta(n)$  (i.e. spectre plat), la pression dans la *bright zone* peut être calculée suivant :

$$\mathbf{p}_B = \mathbf{H}_B \mathbf{w}, \quad (6.7)$$

avec

$$\mathbf{p}_B = \left[ p_B^{(1)}(0) \quad \cdots \quad p_B^{(1)}(2N-1) \quad \cdots \quad p_B^{(M_B)}(0) \quad \cdots \quad p_B^{(M_B)}(2N-1) \right]_{M_B(2N-1)}^\top. \quad (6.8)$$

Et de manière similaire, on peut calculer la pression dans la *dark zone* :

$$\mathbf{p}_D = \mathbf{H}_D \mathbf{w}, \quad (6.9)$$

avec

$$\mathbf{p}_D = \left[ p_D^{(1)}(0) \quad \cdots \quad p_D^{(1)}(2N-1) \quad \cdots \quad p_D^{(M_D)}(0) \quad \cdots \quad p_D^{(M_D)}(2N-1) \right]_{M_D(2N-1)}^\top. \quad (6.10)$$

### 6.4.2 Cas 2 : En prenant en compte le signal d'entrée

Cette fois-ci, le signal d'entrée  $x(n)$  est connu et est utilisé dans les équations. En effet, la présence des microphones dans chaque zone permet de mesurer directement la pression reproduite. Ainsi, au lieu d'écrire les algorithmes de *sound zones* à partir de la pression estimée aux points de contrôle, il est possible de réécrire les algorithmes en fonction des pressions réelles. Notons que classiquement, en exploitation, les microphones sont retirés des zones, car ils peuvent gêner les utilisateurs. Cependant, des techniques existent pour reconstruire les pressions aux points de contrôle à partir de mesures plus éloignées comme les antennes virtuelles [91] ou par reconstruction du champ de pression de la pièce [92].

La pression mesurée par le microphone  $m$  est donnée par l'équation (6.1) qu'il est possible de réécrire par associativité de la convolution :

$$p^{(m)}(n) = \sum_{l=1}^L \left( h^{(m,l)} * w^{(l)} * x \right) (n) = \sum_{l=1}^L \left( w^{(l)} * \left( h^{(m,l)} * x \right) \right) (n). \quad (6.11)$$

Jusqu'à présent, le signal d'entrée n'a pas été pris en compte dans les algorithmes de *sound zones*. Le problème de minimisation de l'erreur quadratique s'appuie sur le calcul de cette erreur dans le cas d'un signal d'entrée à spectre plat (voir section 6.4.1). Pour autant, grâce à la présence des microphones, il est possible de mesurer directement cette erreur et de l'utiliser dans l'algorithme, quel que soit le signal d'entrée  $x(n)$ .

L'équation de convolution qui relie la pression mesurée au microphone  $m$  aux autres grandeurs du problème est :

$$p^{(m)}(n) = \sum_{l=1}^L \sum_{k=0}^{N-1} w^{(l)}(k) \sum_{i=0}^{N-1} h^{(m,l)}(i) x(n-k-i). \quad (6.12)$$

Définissons le signal  $q^{(m,l)}$  qui serait reçu par le microphone  $m$  si un seul haut-parleur  $l$  émettait le signal  $x$  sans filtrage :

$$q^{(m,l)}(n) = \sum_{i=0}^{N-1} h^{(m,l)}(i)x(n-i). \quad (6.13)$$

Par extension, le signal vectoriel  $\mathbf{q}^{(m,l)}(n) \in \mathbb{R}^{N \times 1}$  contenant les  $N$  derniers échantillons du signal  $q^{(m,l)}(n)$  s'écrit :

$$\mathbf{q}^{(m,l)}(n) = \begin{bmatrix} q^{(m,l)}(n) & q^{(m,l)}(n-1) & \dots & q^{(m,l)}(n-N+1) \end{bmatrix}_{N \times 1}^{\top}. \quad (6.14)$$

La pression mesurée par le microphone  $m$  devient alors :

$$p^{(m)}(n) = \mathbf{q}^{(m)\top}(n)\mathbf{w}, \quad (6.15)$$

avec

$$\mathbf{q}^{(m)}(n) = \begin{bmatrix} \mathbf{q}^{(m,1)\top}(n) & \dots & \mathbf{q}^{(m,L)\top}(n) \end{bmatrix}_{NL \times 1}^{\top}. \quad (6.16)$$

Sous forme vectorielle,  $\mathbf{p}_B(n) \in \mathbb{R}^{M_B \times 1}$  le vecteur contenant les pressions de chaque microphone de la *bright zone* s'écrit suivant :

$$\mathbf{p}_B(n) = \mathbf{Q}_B(n)\mathbf{w}, \quad (6.17)$$

avec

$$\mathbf{Q}_B(n) = \begin{bmatrix} \mathbf{q}^{(1)\top}(n) \\ \vdots \\ \mathbf{q}^{(m)\top}(n) \\ \vdots \\ \mathbf{q}^{(M_B)\top}(n) \end{bmatrix}_{M_B \times NL}. \quad (6.18)$$

Enfin, de manière similaire, la pression  $\mathbf{p}_D(n) \in \mathbb{R}^{M_D \times 1}$  aux points de mesure dans la *dark zone* vérifie :

$$\mathbf{p}_D(n) = \mathbf{Q}_D(n)\mathbf{w}, \quad (6.19)$$

avec

$$\mathbf{Q}_D(n) = \begin{bmatrix} \mathbf{q}^{(1)\top}(n) \\ \vdots \\ \mathbf{q}^{(m)\top}(n) \\ \vdots \\ \mathbf{q}^{(M_D)\top}(n) \end{bmatrix}_{M_D \times NL}. \quad (6.20)$$

Les différentes équations régissant le système étant posées, il reste maintenant à définir les contraintes dans les zones de l'espace et c'est l'objet des sections suivantes.

## 6.5 Algorithme BACC

L'algorithme *Acoustic Contrast Control* (ACC) a été présenté pour la première fois en 2002 par CHOI *et al.* [7] dans le domaine fréquentiel. En 2011, ELLIOTT *et al.* proposa la version *Broad-band Acoustic Contrast Control* (BACC) dans le domaine temporel présentée ici. Pour rappel, cet algorithme tente de maximiser le contraste entre les deux zones.

Pour cet algorithme qui maximise le contraste, l'énergie potentielle dans la *bright zone* est calculée par :

$$e_B = \frac{1}{KM_B} \sum_{k=0}^{K-1} \mathbf{p}_B^\top(k) \mathbf{p}_B(k) = \frac{1}{M_B} \mathbf{w}^\top \mathbf{R}_B \mathbf{w}, \quad (6.21)$$

et de manière similaire l'énergie potentielle dans la *dark zone* est calculée par :

$$e_D = \frac{1}{KM_D} \sum_{k=0}^{K-1} \mathbf{p}_D^\top(k) \mathbf{p}_D(k) = \frac{1}{M_D} \mathbf{w}^\top \mathbf{R}_D \mathbf{w}. \quad (6.22)$$

Les matrices  $\mathbf{R}_B \in \mathbb{R}^{LN \times LN}$  et  $\mathbf{R}_D \in \mathbb{R}^{LN \times LN}$  sont des matrices carrées réelles semi-définies positives, dites matrices de corrélation spatiale [63], respectivement de la *bright zone* et de la *dark zone*, définies par :

$$\begin{aligned} \mathbf{R}_B &= \frac{1}{K} \sum_{k=0}^{K-1} \mathbf{Q}_B(k) \mathbf{Q}_B^\top(k), \\ \mathbf{R}_D &= \frac{1}{K} \sum_{k=0}^{K-1} \mathbf{Q}_D(k) \mathbf{Q}_D^\top(k). \end{aligned} \quad (6.23)$$

Le contraste  $\gamma_{\text{BACC}} \in \mathbb{R}$  est alors défini comme le rapport de l'énergie de la *bright zone* sur l'énergie de la *dark zone* [7] :

$$\gamma_{\text{BACC}} = \frac{e_B}{e_D} = \frac{M_D \mathbf{w}^\top \mathbf{R}_B \mathbf{w}}{M_B \mathbf{w}^\top \mathbf{R}_D \mathbf{w}}. \quad (6.24)$$

L'équation précédente peut être réécrite en :

$$\frac{M_D}{M_B} \mathbf{R}_B \mathbf{w} = \gamma_{\text{BACC}} \mathbf{R}_D \mathbf{w}. \quad (6.25)$$

Puisque nous cherchons à obtenir le contraste le plus grand, la solution  $\mathbf{w}_{\text{BACC}}$  à sélectionner correspond au vecteur propre associé à la plus grande valeur propre  $\gamma_{\text{BACC}}^{(\max)}$ .

Tel que présenté ici, l'algorithme BACC ne garantit pas une distribution spatiale de l'énergie dans la *bright zone* uniforme. De plus, la phase n'est pas non plus contrainte dans la *bright zone*.

## 6.6 Algorithme BACC-PM

Pour s'affranchir des problématiques de phase dans la *bright zone*, l'algorithme *Broadband Acoustic Contrast Control with Pressure Matching* (BACC-PM) [59] contraint l'amplitude et la phase dans la *bright zone*, en rajoutant une contrainte sur l'énergie du réseau disponible et l'énergie dans la *dark zone*.

Pour cet algorithme, une pression cible dans la *bright zone* notée  $\mathbf{p}_t$  doit être au préalable définie. L'algorithme présenté ici est formulé comme dans la publication originale [59], sans prendre en compte le signal d'entrée.

Nous souhaitons minimiser l'erreur de reproduction dans la *bright zone*. Pour parvenir à cet objectif, nous sommes cependant limités par l'énergie maximale  $E_0$  que peut reproduire le réseau. De manière complémentaire, nous souhaitons limiter autant que possible la pression dans la *dark zone* à un niveau défini  $D_0$ . Ainsi, le problème d'optimisation peut

se poser sous la forme suivante :

$$\left\{ \begin{array}{l} \text{minimiser } \|\mathbf{H}_B \mathbf{w} - \mathbf{p}_t\|_2^2 \quad \text{par rapport à } \mathbf{w} \\ \text{contraint en principal par } \|\mathbf{w}\|_2^2 < E_0 \\ \text{et en secondaire par } \|\mathbf{H}_D \mathbf{w}\|_2^2 < D_0 \end{array} \right. . \quad (6.26)$$

On construit alors le Lagrangien suivant,  $(\lambda_1, \lambda_2) \in \mathbb{R}_+^2$  :

$$J(\mathbf{w}) = \|\mathbf{H}_B \mathbf{w} - \mathbf{p}_t\|_2^2 + \lambda_1(\|\mathbf{H}_D \mathbf{w}\|_2^2 - D_0) + \lambda_2(\|\mathbf{w}\|_2^2 - E_0). \quad (6.27)$$

Dérivant l'équation précédente par rapport à  $\mathbf{w}$ , on obtient :

$$\frac{\partial J}{\partial \mathbf{w}} = 2\mathbf{H}_B^\top \mathbf{H}_B \mathbf{w} - 2\mathbf{H}_B^\top \mathbf{p}_t + 2\lambda_1 \mathbf{H}_D^\top \mathbf{H}_D \mathbf{w} + 2\lambda_2 \mathbf{w}. \quad (6.28)$$

Puis, la dérivée par rapport aux deux multiplicateurs donne :

$$\left\{ \begin{array}{l} \frac{\partial J}{\partial \lambda_1} = \|\mathbf{H}_D \mathbf{w}\|_2^2 - D_0 = 0 \quad \Rightarrow \quad \|\mathbf{H}_D \mathbf{w}\|_2^2 = D_0 \\ \frac{\partial J}{\partial \lambda_2} = \|\mathbf{w}\|_2^2 - E_0 = 0 \quad \Rightarrow \quad \|\mathbf{w}\|_2^2 = E_0 \end{array} \right. . \quad (6.29)$$

Un point stationnaire est atteint lorsque cette dérivée s'annule, soit pour :

$$\mathbf{w} = [\mathbf{H}_B^\top \mathbf{H}_B + \lambda_1 \mathbf{H}_D^\top \mathbf{H}_D + \lambda_2 \mathbf{I}]^{-1} \mathbf{H}_B^\top \mathbf{p}_t. \quad (6.30)$$

Pour résoudre le problème d'optimisation complet avec les contraintes, il est par exemple possible d'utiliser les conditions de KARUSH-KUHN-TUCKER [93], ou alors, de manière empirique, de s'appuyer sur les connaissances physiques que nous avons du problème. C'est le second choix qui est retenu ici.

Le Lagrangien est une formulation énergétique du problème, ainsi le premier terme de la solution (6.30) représente l'énergie dans la *bright zone*, le second terme l'énergie dans la *dark zone* et le dernier terme l'énergie du réseau.

Lors des simulations, l'énergie du réseau n'est pas contrainte, mais peut par contre être mesurée. Ainsi, nous ne considérerons pas la contrainte d'énergie du réseau :  $\lambda_2$  n'est donc plus une contrainte, mais un paramètre de régularisation, noté  $\lambda$  par la suite. Il sera choisi le plus petit possible tout en veillant à ce que le conditionnement de la matrice à inverser



reste correct.

Pour le choix des valeurs des multiplicateurs de Lagrange, une solution a été apportée par CHANG *et al.* [94], dans le domaine fréquentiel, dite méthode combinée, puis appliquée au cas temporel par SIMON-GALVEZ *et al.* [59]. Un paramètre  $\beta \in [0, 1]$  est ajouté à l’algorithme, dont l’utilisateur peut modifier la valeur, qui permet de choisir entre le meilleur contraste et la plus petite erreur.

On obtient finalement :

$$\mathbf{w}_{\text{BACC-CPM}} = \left[ (1 - \beta) \mathbf{H}_B^\top \mathbf{H}_B + \beta \mathbf{H}_D^\top \mathbf{H}_D + \lambda \mathbf{I} \right]^{-1} (1 - \beta) \mathbf{H}_B^\top \mathbf{p}_t. \quad (6.31)$$

La solution précédente nécessite l’inversion d’une matrice, qui est de dimension  $NL \times NL$  (nombre de coefficients du filtre par le nombre de haut-parleurs). Or le calcul de l’inverse d’une matrice a un coût de calcul en  $O(n^3/3)$ , l’inversion de cette matrice (de plusieurs milliers de points) peut devenir prohibitif s’il faut l’inverser à chaque instant. De plus, l’inversion directe à chaque instant ne garantit pas d’avoir des changements minimes des filtres, ce qui provoquerait des artefacts dans le contenu audio reproduit.

## 6.7 Conclusion

Les formulations actuelles des algorithmes de *sound zones* ne permettent pas une mise à jour au cours du temps des filtres en fonction de l’évolution des réponses impulsionnelles qui caractérisent le système. Dans le chapitre suivant, nous partirons de la solution du BACC-PM pour la rendre récursive.

# VERSION RÉCURSIVE DE L'ALGORITHME BACC-PM

---

Pour pouvoir résoudre le problème de *sound zones* dans un cas variant dans le temps, nous allons développer dans ce chapitre une méthode de résolution récursive. Ainsi, lorsque les réponses impulsionnelles changeront, l'algorithme prendra en compte les changements pour trouver une nouvelle solution, au cours du temps.

De plus, la méthode présentée permettra de s'affranchir de la résolution d'un problème aux valeurs propres ou d'une inversion de matrice de grande dimension.

Pour cela, nous partirons de l'équation (6.31), solution explicite BACC-PM, pour la rendre récursive.

## 7.1 Résolution par descente de gradient

De la même manière que pour le filtre LMS, nous proposons ici une solution récursive au problème d'optimisation (6.31). Soit  $\mathbf{w}_k$  le vecteur poids à l'itération  $k$ . Nous cherchons donc une meilleure solution par descente de gradient à l'instant suivant sous la forme :

$$\mathbf{w}_{k+1} = \mathbf{w}_k - \frac{1}{2}\theta_k \nabla J(\mathbf{w})|_{\mathbf{w}=\mathbf{w}_k}, \quad (7.1)$$

avec  $\theta_k$  le pas de descente du gradient, pouvant être dépendant du temps.

Le Lagrangien associé au problème à solution combinée s'écrit :

$$\begin{aligned} J(\mathbf{w}) &= (1 - \beta)\|\mathbf{H}_B\mathbf{w} - \mathbf{p}_t\|_2^2 + \beta(\|\mathbf{H}_D\mathbf{w}\|_2^2) + \lambda(\|\mathbf{w}\|_2^2) \\ &= (1 - \beta) \left[ \mathbf{w}^\top \mathbf{H}_B^\top \mathbf{H}_B \mathbf{w} - 2\mathbf{w}^\top \mathbf{H}_B^\top \mathbf{p}_t + \mathbf{p}_t^\top \mathbf{p}_t \right] + \beta \mathbf{w}^\top \mathbf{H}_D^\top \mathbf{H}_D \mathbf{w} + \lambda \mathbf{w}^\top \mathbf{w} \end{aligned} \quad (7.2)$$

Notant par la suite :

$$\mathbf{H}_T = (1 - \beta)\mathbf{H}_B^\top \mathbf{H}_B + \beta\mathbf{H}_D^\top \mathbf{H}_D + \lambda\mathbf{I}, \quad (7.3)$$

le gradient  $\nabla J(\mathbf{w})$  de la fonction de coût précédente à l'instant  $k$  s'écrit :

$$\nabla J(\mathbf{w})|_{\mathbf{w}=\mathbf{w}_k} = 2\mathbf{H}_T \mathbf{w}_k - 2(1 - \beta)\mathbf{H}_B^\top \mathbf{p}_t, \quad (7.4)$$

soit

$$\mathbf{w}_{k+1} = \mathbf{w}_k - \theta_k \left( \mathbf{H}_T \mathbf{w}_k - (1 - \beta)\mathbf{H}_B^\top \mathbf{p}_t \right). \quad (7.5)$$

Afin d'alléger les notations, nous notons enfin :

$$\mathbf{r}_k = -\frac{1}{2} \nabla J(\mathbf{w})|_{\mathbf{w}=\mathbf{w}_k}. \quad (7.6)$$

Pour calculer le pas de descente de gradient  $\theta_k$ , nous souhaitons que la variation de  $\mathbf{w}_k + \theta_k \mathbf{r}_k$  par rapport à  $\theta_k$  soit minimale, afin que les artefacts audios soient limités.

Calculons alors la dérivée de la fonction coût  $J(\mathbf{w}_k + \theta_k \mathbf{r}_k)$  par rapport à  $\theta_k$  en détaillant les différentes étapes qui mènent à la solution :

$$\begin{aligned} \frac{dJ(\mathbf{w}_k + \theta_k \mathbf{r}_k)}{d\theta_k} &= \nabla J(\mathbf{w}_k + \theta_k \mathbf{r}_k) \frac{d}{d\theta_k} (\mathbf{w}_k + \theta_k \mathbf{r}_k) \\ &= \left( 2\mathbf{H}_T (\mathbf{w}_k + \theta_k \mathbf{r}_k) - 2(1 - \beta)\mathbf{H}_B^\top \mathbf{p}_t \right)^\top \mathbf{r}_k \\ &= \left( 2\mathbf{H}_T \mathbf{w}_k + 2\theta_k \mathbf{H}_T \mathbf{r}_k - 2(1 - \beta)\mathbf{H}_B^\top \mathbf{p}_t \right)^\top \mathbf{r}_k \\ &= (2\theta_k \mathbf{H}_T \mathbf{r}_k - 2\mathbf{r}_k)^\top \mathbf{r}_k \\ &= \left( 2\theta_k (\mathbf{H}_T \mathbf{r}_k)^\top - 2\mathbf{r}_k^\top \right) \mathbf{r}_k \\ &= 2\theta_k (\mathbf{H}_T \mathbf{r}_k)^\top \mathbf{r}_k - 2\mathbf{r}_k^\top \mathbf{r}_k \\ &= 2\theta_k \mathbf{r}_k^\top \mathbf{H}_T^\top \mathbf{r}_k - 2\mathbf{r}_k^\top \mathbf{r}_k. \end{aligned} \quad (7.7)$$

Le minimum est atteint lorsque la dérivée s'annule, donc pour :

$$\theta_k = \frac{\mathbf{r}_k^\top \mathbf{r}_k}{\mathbf{r}_k^\top \mathbf{H}_T^\top \mathbf{r}_k}. \quad (7.8)$$

Finalement, l'équation de récurrence devient :

$$\mathbf{w}_{k+1} = \mathbf{w}_k - \frac{\mathbf{r}_k^\top \mathbf{r}_k}{\mathbf{r}_k^\top \mathbf{H}_T \mathbf{r}_k} \left( \mathbf{H}_T \mathbf{w}_k - (1 - \beta) \mathbf{H}_B^\top \mathbf{p}_t \right). \quad (7.9)$$

L'équation de récurrence précédente permet alors de calculer les filtres de *sound zones* au cours du temps, chaque pas de descente de gradient  $k$  étant confondu avec le temps  $n$ . Il reste à prouver que l'équation de récurrence permette de trouver la même solution que la solution explicite, ce qui fait l'objet de la section suivante.

### 7.1.1 Étude de convergence

Les matrices  $\mathbf{H}_B$  et  $\mathbf{H}_D$  sont supposées non dégénérées, sinon cela reviendrait physiquement à dire qu'il y a deux réponses impulsionnelles identiques. Les matrices  $\mathbf{H}_B^\top \mathbf{H}_B$  et  $\mathbf{H}_D^\top \mathbf{H}_D$  sont donc définies positives, la matrice identité étant également définie positive. Or la somme de matrices définies positives est une matrice définie positive, ainsi la matrice carrée  $\mathbf{H}_T$  (équation (7.3)) est définie positive [95].

Définissons le vecteur erreur  $\boldsymbol{\epsilon}$  comme la différence entre l'estimation de vecteur poids  $\mathbf{w}$  et la solution  $\mathbf{w}_o$  :

$$\boldsymbol{\epsilon} = \mathbf{w} - \mathbf{w}_o. \quad (7.10)$$

Notons alors la fonction coût  $J_2(\mathbf{w})$  :

$$J_2(\mathbf{w}) = \frac{1}{2} \boldsymbol{\epsilon}^\top \mathbf{H}_T \boldsymbol{\epsilon} = \frac{1}{2} (\mathbf{w} - \mathbf{w}_o)^\top \mathbf{H}_T (\mathbf{w} - \mathbf{w}_o) = J(\mathbf{w}) + \frac{1}{2} \mathbf{w}_o^\top \mathbf{H}_T \mathbf{w}_o. \quad (7.11)$$

Minimiser  $J_2$  revient à minimiser  $J$  puisque les deux fonctions sont égales à une constante près. De plus, le gradient de la fonction  $J$  est identique au gradient de la fonction  $J_2$  [96].

Nous avons alors :

$$\frac{J_2(\mathbf{w}_k) - J_2(\mathbf{w}_{k+1})}{J_2(\mathbf{w}_k)} = \frac{(\mathbf{r}_k^\top \mathbf{r}_k)^2}{(\mathbf{r}_k^\top \mathbf{H}_T \mathbf{r}_k)(\mathbf{r}_k^\top \mathbf{H}_T^{-1} \mathbf{r}_k)}, \quad (7.12)$$

soit :

$$J_2(\mathbf{w}_{k+1}) = \left( 1 - \frac{(\mathbf{r}_k^\top \mathbf{r}_k)^2}{(\mathbf{r}_k^\top \mathbf{H}_T \mathbf{r}_k)(\mathbf{r}_k^\top \mathbf{H}_T^{-1} \mathbf{r}_k)} \right) J_2(\mathbf{w}_k). \quad (7.13)$$

En appliquant l'inégalité de KANTOROVITCH à l'équation précédente [97], nous obtenons :

$$J_2(\mathbf{w}_{k+1}) \leq \left( \frac{\lambda_{\max} - \lambda_{\min}}{\lambda_{\max} + \lambda_{\min}} \right)^2 J_2(\mathbf{w}_k), \quad (7.14)$$

avec  $\lambda_{\max}$  et  $\lambda_{\min}$  les valeurs propres maximale et minimale de la matrice  $\mathbf{H}_T$ . Or celle-ci est définie positive et non dégénérée, donc :

$$J_2(\mathbf{w}_{k+1}) < J_2(\mathbf{w}_k) \Rightarrow J_2(\mathbf{w}_k) \rightarrow 0. \quad (7.15)$$

Puisque la matrice  $\mathbf{H}_T$  est définie positive et non dégénérée, et puisque  $J_2$  tend vers 0, alors  $\mathbf{w}_k \rightarrow \mathbf{w}_o$ .

Cette formulation du problème d'optimisation de l'algorithme de *pressure matching* BACC-PM permet alors de résoudre le problème au cours du temps, sans ajout de paramètre supplémentaire. Il sera nommé par la suite BACC-PM-Gd.

### 7.1.2 Gradient conjugué

Une descente de gradient simple souffre d'une dépendance de sa vitesse de convergence en fonction du conditionnement de la matrice  $\mathbf{H}_T$ . Pour accélérer la convergence, une méthode usuelle consiste à utiliser une descente de gradients conjugués [96]. C'est par exemple cette méthode d'optimisation qui est choisie par SHI *et al.* [83] pour résoudre le problème, dans le cas de *sound zones* statiques, le problème en question étant formulé dans le domaine fréquentiel.

Pour rappel, le système à résoudre dans notre cas s'écrit :

$$\mathbf{H}_T \mathbf{w} = (1 - \beta) \mathbf{H}_B^\top \mathbf{p}_t, \quad (7.16)$$

avec la matrice  $\mathbf{H}_T$  de taille  $NL \times NL$ .

L'unique solution  $\mathbf{w}_o$  de l'équation précédente est aussi l'unique solution du problème de minimisation quadratique [96] :

$$\min_{\mathbf{w}} \left( \frac{1}{2} \mathbf{w}^\top \mathbf{H}_T \mathbf{w} - \left( (1 - \beta) \mathbf{H}_B^\top \mathbf{p}_t \right)^\top \mathbf{w} \right), \mathbf{w} \in \mathbb{R}^{NL}. \quad (7.17)$$

Soient les vecteurs  $\boldsymbol{\omega}_0, \boldsymbol{\omega}_1, \dots, \boldsymbol{\omega}_{NL-1}$  non nuls et  $\mathbf{H}_T$ -orthogonaux<sup>1</sup>, donc linéairement indépendants. La solution unique  $\mathbf{w}_o$  peut s'écrire [96] :

$$\mathbf{w}_o = \sum_{i=0}^{NL-1} \alpha_i \boldsymbol{\omega}_i, \quad (7.18)$$

avec  $\alpha_i$  les coefficients à calculer.

On multiplie alors l'équation précédente à gauche par  $\mathbf{H}_T$ , puis on prend le produit scalaire par  $\boldsymbol{\omega}_i$  et, en utilisant la propriété de  $\mathbf{H}_T$ -orthogonalité, on obtient :

$$\alpha_i = \frac{\boldsymbol{\omega}_i^\top \mathbf{H}_T \mathbf{w}_o}{\boldsymbol{\omega}_i^\top \mathbf{H}_T \boldsymbol{\omega}_i} = (1 - \beta) \frac{\boldsymbol{\omega}_i \mathbf{H}_B^\top \boldsymbol{\omega}_t}{\boldsymbol{\omega}_i^\top \mathbf{H}_T \boldsymbol{\omega}_i}. \quad (7.19)$$

Le fait de choisir une base de vecteurs  $\mathbf{H}_T$ -orthogonaux permet de limiter la somme de l'équation (7.18) à  $NL$  termes et de pouvoir calculer les coefficients  $\alpha_i$  par l'équation (7.19) sans avoir besoin de la solution  $\mathbf{w}_o$ .

Il reste maintenant à calculer la base des vecteurs  $\boldsymbol{\omega}_i$  et les coefficients associés  $\alpha_i$  de manière récursive.

La somme (7.18) peut être calculée récursivement suivant :

$$\mathbf{w}_{k+1} = \mathbf{w}_k + \alpha_k \boldsymbol{\omega}_k, \quad (7.20)$$

qui est également l'équation de mise à jour des filtres de *sound zones*.

Or, d'après l'équation précédente, à l'itération finale  $NL$  et avec une condition initiale  $\mathbf{w}_0$ , nous avons :

$$\mathbf{w}_o - \mathbf{w}_0 = \alpha_0 \boldsymbol{\omega}_0 + \dots + \alpha_k \boldsymbol{\omega}_k + \dots + \alpha_{NL-1} \boldsymbol{\omega}_{NL-1}. \quad (7.21)$$

En multipliant alors l'équation précédente à gauche par  $\mathbf{H}_T$  et en prenant le produit scalaire par  $\boldsymbol{\omega}_k$ , on obtient :

$$\boldsymbol{\omega}_k \mathbf{H}_T (\mathbf{w}_o - \mathbf{w}_0) = \alpha_k \boldsymbol{\omega}_k^\top \mathbf{H}_T \boldsymbol{\omega}_k, \quad (7.22)$$

---

1. Deux vecteurs  $\boldsymbol{\omega}_0$  et  $\boldsymbol{\omega}_1$  sont  $\mathbf{H}_T$ -orthogonaux si  $\boldsymbol{\omega}_0^\top \mathbf{H}_T \boldsymbol{\omega}_1 = 0$ .

soit

$$\alpha_k = \frac{\boldsymbol{\omega}_k^\top \mathbf{H}_T (\mathbf{w}_o - \mathbf{w}_0)}{\boldsymbol{\omega}_k^\top \mathbf{H}_T \boldsymbol{\omega}_k}. \quad (7.23)$$

De plus, en s'arrêtant à l'itération  $k$  et en effectuant le même calcul que précédemment sur la somme, nous trouvons :

$$\boldsymbol{\omega}_k^\top \mathbf{H}_T (\mathbf{w}_k - \mathbf{w}_0) = 0, \quad (7.24)$$

soit

$$\boldsymbol{\omega}_k^\top \mathbf{H}_T \mathbf{w}_k = \boldsymbol{\omega}_k^\top \mathbf{H}_T \mathbf{w}_0. \quad (7.25)$$

Il est donc possible de remplacer  $\mathbf{w}_o - \mathbf{w}_0$  par  $\mathbf{w}_o - \mathbf{w}_k$  dans l'équation (7.23). Nous savons de plus que  $\mathbf{w}_o = (1 - \beta) \mathbf{H}_T^{-1} \mathbf{H}_B \boldsymbol{\omega}_t$  et en prenant le vecteur  $\mathbf{r}_k$  défini à l'équation (7.6), nous obtenons finalement :

$$\begin{aligned} \alpha_k &= \frac{\boldsymbol{\omega}_k^\top \mathbf{H}_T \left( (1 - \beta) \mathbf{H}_T^{-1} \mathbf{H}_B \boldsymbol{\omega}_t - \mathbf{w}_k \right)}{\boldsymbol{\omega}_k^\top \mathbf{H}_T \boldsymbol{\omega}_k} \\ &= \frac{\boldsymbol{\omega}_k^\top \left( (1 - \beta) \mathbf{H}_B^\top \boldsymbol{\omega}_t - \mathbf{H}_T \mathbf{w}_k \right)}{\boldsymbol{\omega}_k^\top \mathbf{H}_T \boldsymbol{\omega}_k} \\ &= -\frac{\mathbf{r}_k^\top \boldsymbol{\omega}_k}{\boldsymbol{\omega}_k^\top \mathbf{H}_T \boldsymbol{\omega}_k}. \end{aligned} \quad (7.26)$$

C'est cette équation qui nous permet alors de calculer le paramètre  $\alpha_k$  dans l'algorithme.

Il ne nous reste plus qu'à définir les vecteurs  $\mathbf{H}_T$ -orthogonaux  $\boldsymbol{\omega}_k$  qui représentent la direction du gradient à l'itération  $k + 1$ , qui est une combinaison du gradient actuel avec les directions passées [96] :

$$\boldsymbol{\omega}_{k+1} = \mathbf{r}_k + \chi_k \boldsymbol{\omega}_k, \quad (7.27)$$

avec  $\chi_k$  le coefficient qui assure l' $\mathbf{H}_T$ -orthogonalité des vecteurs [96] :

$$\chi_k = \frac{\mathbf{r}_{k+1}^\top \mathbf{H}_T \boldsymbol{\omega}_k}{\boldsymbol{\omega}_k^\top \mathbf{H}_T \boldsymbol{\omega}_k}. \quad (7.28)$$

Pour effectuer une descente du gradient conjugué, il faut alors exécuter dans l'ordre les équations (7.20), (7.26), (7.28) et (7.27).

Le gradient conjugué permet, en contexte statique, d'accélérer la convergence par rapport à la descente de gradient classique. De plus, il permet de connaître en avance le temps maximal de convergence puisqu'il est prouvé qu'il converge au maximum en  $NL$  itérations [98]. Cependant, dans le cas variant dans le temps, la surface d'erreur change au cours du temps. La base de vecteurs conjugués perd donc sa caractéristique d'orthogonalité, c'est-à-dire qu'une fois une direction explorée, elle n'est plus analysée : ainsi, si la surface d'erreur change au cours du temps, certaines directions risquent de ne pas être explorées. Pour autant, des stratégies peuvent être mises en place, par exemple en réinitialisant la base en cas de changement.

## 7.2 Simulations en contexte statique

Dans cette partie, les algorithmes de descente de gradient (BACC-PM-Gd) et de descente de gradient conjugué (BACC-PM-Gdc) vont être testés dans le cadre de simulations numériques simples pour confirmer les résultats théoriques présentés précédemment.

### 7.2.1 Géométrie

La première simulation s'effectue en champ libre avec un écho (salle semi-anéchoïque). Douze sources monopolaires sont disposées comme indiqué en figure 7.1. Les sources et les microphones sont placés à un mètre de hauteur. La *bright zone* (bleu) est échantillonnée par une antenne de neuf microphones centrée en  $(0.6, -0.6)$  et la *dark zone* (orange) est échantillonnée par une antenne de neuf microphones centrée en  $(2.4, -0.6)$ . L'espacement entre deux microphones voisins est de 0,1 m.

Les réponses impulsionnelles à temps continu peuvent être modélisées analytiquement par [43] :

$$h^{(l,m)}(t) = \alpha_1^{(l,m)} \delta(t - \Delta t_1^{(l,m)}) + \alpha_2^{(l,m)} \delta(t - \Delta t_2^{(l,m)}) \quad (7.29)$$

avec  $\alpha_{1,2}^{(l,m)} = 1/4\pi d_{1,2}^{(l,m)}$  les coefficients d'atténuation des chemins direct (1) et réfléchi (2) et  $\Delta t_{1,2}^{(l,m)} = d_{1,2}^{(l,m)}/c_0$  les temps de vol correspondant aux distances  $d_{1,2}^{(l,m)}$  entre haut-parleur  $l$  et microphone  $m$ ,  $c_0$  étant la célérité des ondes acoustiques dans l'air.

La mesure est quant à elle effectuée dans la salle semi-anéchoïque du LAUM (voir figure 7.2). Les réponses impulsionnelles sont mesurées par la méthode ESS (section 2.1). Le dispositif expérimental est contrôlé par une interface audio RME Madiface XT qui



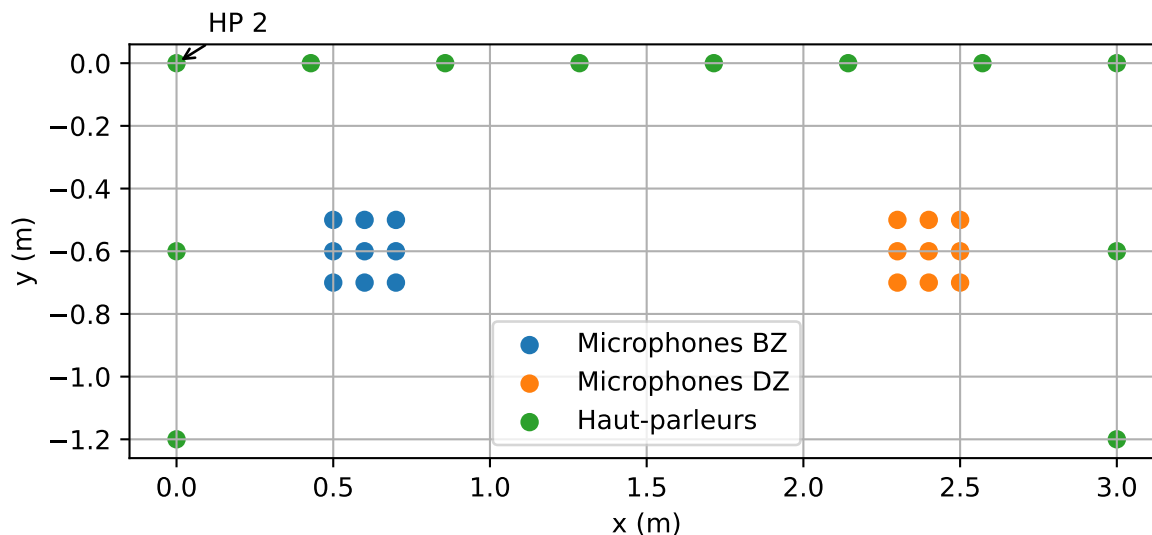


FIGURE 7.1 – Géométrie considérée dans ce chapitre.

permet la synchronisation de l’acquisition et la reproduction. Les sources audio utilisées sont des enceintes à évent deux voies fabriquées au LAUM. Elles sont alimentées par des signaux issus d’un convertisseur RME DA32 qui sont amplifiés par des modules de puissance quatre voies HPA D604. Le champ de pression résultant est mesuré par dix-huit microphones Brüel & Kjør 4958 connectés à leurs conditionneurs puis numérisé par un convertisseur RME AD32. L’acquisition et la reproduction sont effectuées à une fréquence d’échantillonnage de 48 kHz. Les réponses impulsionnelles mesurées sont alors utilisées dans les algorithmes, les métriques étant évaluées *off-line*.

### 7.2.2 Métriques

Pour analyser les performances de l’algorithme, il est nécessaire de définir un certain nombre d’indicateurs. Ces indicateurs seront analysés dans un premier temps en contexte

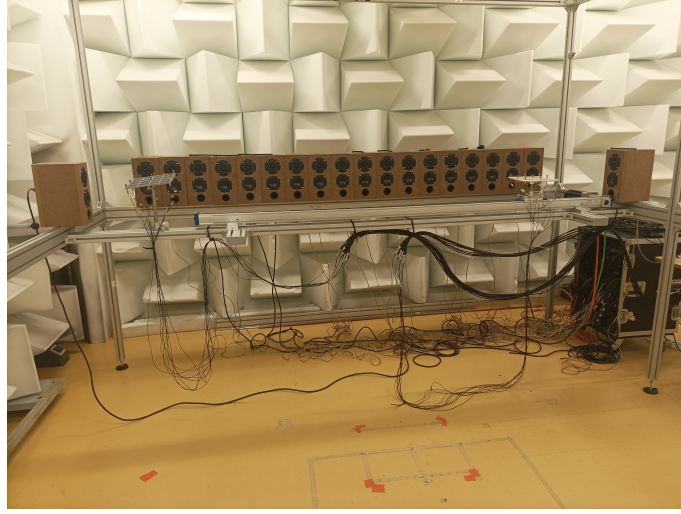


FIGURE 7.2 – Dispositif expérimental installé dans la salle semi-anéchoïque.

statique, puis dans un second temps pour diverses situations dynamiques. La définition du contraste est reprise de ELLIOTT [10] afin de pouvoir comparer les résultats de l'algorithme à ceux de l'état de l'art.

**Pression RMS** A l'instant  $n$ , on définit le niveau de pression RMS en Pascal en un point, tel que :

$$p_{\text{RMS}}(n) = \sqrt{\frac{1}{N_w} \sum_{m=n-(N_w-1)}^n p^2(m)}, \quad (7.30)$$

avec  $N_w$  le nombre de points correspondant au temps d'intégration de 0,1 s.

**Erreur de pression** Cet indicateur code pour chaque instant  $n$  la capacité de l'algorithme à atteindre la pression cible demandée et est défini suivant :

$$\text{EP}(n) = \frac{1}{M} \sum_{m=1}^M |p_B^{(m)}(n) - p_t^{(m)}(n)|. \quad (7.31)$$

**Contraste** Le contraste est le rapport (exprimé en dB) entre les pressions RMS dans la *bright zone* et dans la *dark zone* :

$$\text{CT}(n) = 20 \log_{10} \left( \frac{1}{M} \sum_{m=1}^M p_{\text{RMS},B}^{(m)}(n) \right) - 20 \log_{10} \left( \frac{1}{M} \sum_{m=1}^M p_{\text{RMS},D}^{(m)}(n) \right). \quad (7.32)$$

**Effort du réseau** L’effort du réseau mesure l’énergie nécessaire à fournir par les haut-parleurs pour atteindre les performances souhaitées. Dans le cadre des simulations, cet indicateur est mesuré a posteriori. Dans le cadre d’expériences, l’effort est limité par le système. On le définit suivant :

$$\text{EFF} = \frac{\sum_{l=1}^L \sqrt{\frac{1}{N} \sum_{k=1}^N (w^{(l)}(k))^2}}{\sqrt{\frac{1}{N} \sum_{k=1}^N (w^{(\text{target})}(k))^2}}. \quad (7.33)$$

L’équation (7.33) définit l’effort du réseau supplémentaire à fournir pour créer les deux zones, par rapport à la pression que pourrait générer le haut-parleur pour créer la pression cible dans la *bright zone*.

### 7.2.3 Résultats

Dans cette partie, nous présenterons des résultats de simulation après 2000 itérations, après 100 itérations et enfin avec des réponses impulsionnelles mesurées.

L’algorithme BACC-PM-Gd ne nécessite pas de paramètre supplémentaire de réglage par rapport à l’algorithme BACC-PM classique. Cependant, il lui faut un certain temps de convergence avant de tendre vers la solution.

Une méthode de résolution par descente de gradient souffre d’une limitation importante : la vitesse de descente de gradient dépend du conditionnement de la matrice  $\mathbf{H}_T$ . Or, dans le cas des *sound zones*, cette matrice peut être mal conditionnée, et la problématique de la régularisation a fait l’objet de nombreuses publications [78]. Donc, dans le cas d’un problème mal posé, la convergence de cet algorithme peut être longue et non bornée.

Pour s’affranchir de cette limitation et, a minima, s’assurer d’un temps minimal de convergence, l’algorithme de descente du gradient conjugué peut être utilisé. Cette technique a alors l’avantage de pouvoir borner le temps de convergence puisque l’algorithme converge au maximum en  $NL$  itérations [98],  $NL$  étant la dimension de la matrice (carrée)  $\mathbf{H}_T$ .

Concernant le temps de calcul, la descente du gradient conjugué nécessite 3,7 fois plus de temps de calcul que la descente simple<sup>2</sup>.

Pour étudier la convergence des algorithmes, nous reprenons la même métrique que dans la première partie de cette thèse, équation (3.18) :

$$\text{mis}(n) = \|\mathbf{w}_{\text{BACC-PM}} - \mathbf{w}_{\text{BACC-PM-Gd}}(n)\|_2. \quad (7.34)$$

La figure 7.3 présente les courbes de convergence de l’algorithme BACC-PM-Gd et de l’algorithme BACC-PM-Gdc. Pour rappel, la taille des filtres est de 300 points et comme il y a 12 haut-parleurs, la taille de la matrice  $\mathbf{H}_T$  est donc  $3600 \times 3600$ . Nous observons une convergence bien plus rapide pour le gradient conjugué par rapport à un gradient simple. De plus, le temps de convergence du gradient conjugué est très inférieur à 3600 itérations.

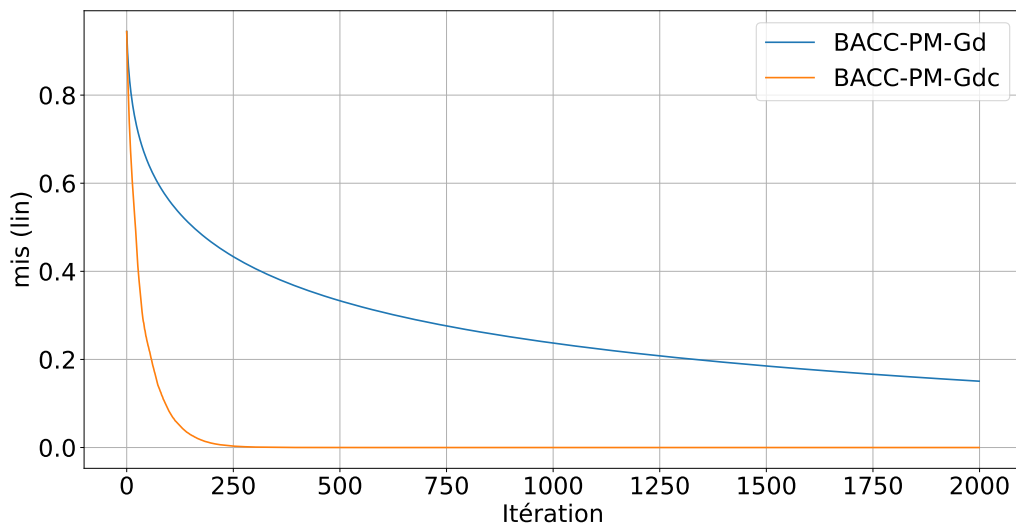


FIGURE 7.3 – Erreur de convergence BACC-PM-Gd et BACC-PM-Gdc.

À la fin de la convergence, nous obtenons les contrastes de la figure 7.4. L’algorithme BACC-PM-Gdc ayant une erreur quasiment nulle à la fin, il est normal que la courbe

2. Calcul effectué sur Intel i7-10875H @ 2,3 GHz | 64 Gb RAM sur 5000 itérations.

de contraste correspond exactement au contraste de l’algorithme BACC-PM. Pour l’algorithme BACC-PM-Gd, le contraste est un peu moins bon en-dessous de 600 Hz, jusqu’à 5 dB inférieur à BACC-PM à 400 Hz. Au-delà de 600 Hz, les contrastes des trois algorithmes sont identiques. À la fin de la convergence, les erreurs de reproduction sont tracées sur la figure 7.5. Nous notons pour l’algorithme BACC-PM une erreur comprise entre  $1 \times 10^{-7}$  et  $1 \times 10^{-5}$ , ce qui est comparable à la littérature [7], [9], [16]. Nous notons également une erreur croissante avec la fréquence, comportement ici aussi classique. Pour les deux algorithmes de descente de gradient, l’erreur de reproduction s’établit à  $1 \times 10^{-6}$  et est quasiment identique, et constante en fréquence.

Un exemple de filtre est donné sur la figure 7.6 (haut-parleur n°0), obtenu avec les algorithmes BACC-PM, BACC-PM-Gd et BACC-PM-Gdc. Les courbes bleue (BACC-PM) et verte (BACC-PM-Gdc) sont confondues, ce qui confirme que l’algorithme BACC-PM-Gdc a trouvé la même solution que la solution explicite. L’effort du réseau est affiché sur la table 7.1, les valeurs pour l’algorithme BACC-PM est identique à l’algorithme BACC-PM-Gdc. Pour l’algorithme BACC-PM-Gd l’effort est de 0,5 dB supérieur, cependant les valeurs sont physiquement possibles et proches de celles de la littérature.

	BACC-PM	BACC-PM-Gd	BACC-PM-Gdc
EFF (dB)	7,4	7,9	7,4

TABLE 7.1 – Effort du réseau des méthodes BACC-PM, BACC-PM-Gd et BACC-PM-Gdc après 2000 itérations.

Au lieu d’attendre la fin de la convergence, nous nous intéressons maintenant à des temps très courts, car, en exploitation, la contrainte de temps est forte pour l’utilisation (il convient en effet d’éviter d’attendre longtemps avant d’avoir un système utilisable). Nous arrêtons la descente de gradient après 100 itérations (soit 0,03 s). L’erreur relative est alors de 55 % pour l’algorithme BACC-PM-Gd et de 10 % pour l’algorithme BACC-PM-Gdc. Nous obtenons les courbes de contraste de la figure 7.7 et d’erreur de la figure 7.8. La courbe de contraste de l’algorithme BACC-PM-Gdc est très proche de la solution explicite, hormis sur la bande 250 Hz-500 Hz où le contraste est plus bas de 2 dB en faveur de la solution explicite. Quant à l’erreur de reproduction, les courbes de la solution explicite et de la descente de gradient conjugué sont quasiment superposées. Pour la descente de gradient simple, les résultats sont moins bons, avec un contraste pouvant être inférieur

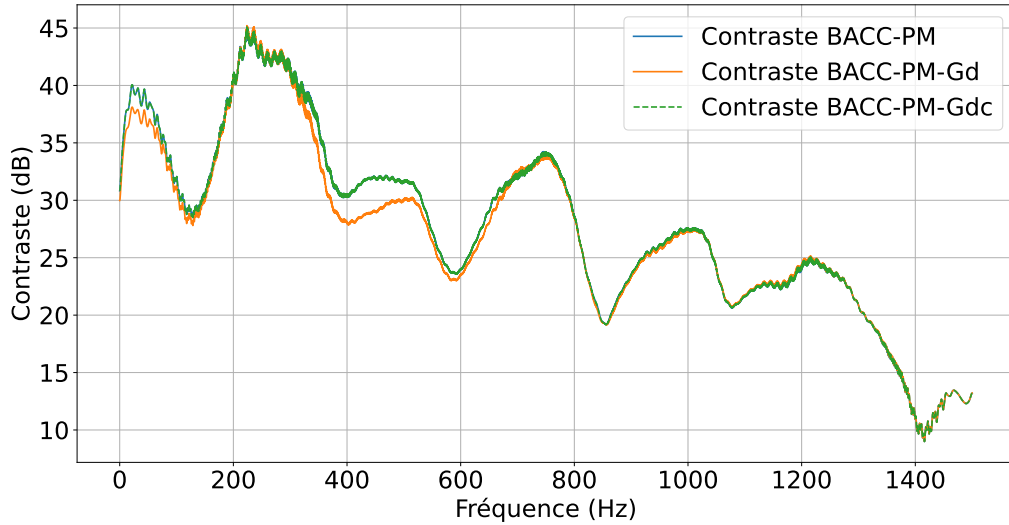


FIGURE 7.4 – Contraste en fonction de la fréquence des méthodes BACC-PM, BACC-PM-Gd et BACC-PM-Gdc après 2000 itérations. Les courbes BACC-PM et BACC-PM-Gdc sont superposées.

de 7 dB en dessous de 600 Hz et une erreur relative un ordre de grandeur plus grande. Pour autant, le contraste reste autour de 25 dB et l'erreur de reproduction de l'ordre de  $1 \times 10^{-6}$ , ce qui est proche du cahier des performances désirées, et ce, dans un temps très court. L'effort du réseau est présenté table 7.2. Nous remarquons que l'algorithme BACC-PM-Gd a un effort bien plus faible, et au vu de l'erreur de reproduction, cela tendrait à montrer que la *dark zone* n'a pas encore été correctement créée.

	BACC-PM	BACC-PM-Gd	BACC-PM-Gdc
EFF (dB)	7,9	4,6	7,8

TABLE 7.2 – Effort du réseau des méthodes BACC-PM, BACC-PM-Gd et BACC-PM-Gdc après 100 itérations.

Concernant la mesure, les courbes de convergence des algorithmes sont présentées en figure 7.9. À nouveau, l'algorithme BACC-PM-Gdc converge plus rapidement que l'algorithme BACC-PM-Gd, et en moins de  $NL = 3600$  itérations. Avec des réponses impulsionnelles réelles, le système est plus complexe (et donc la matrice  $\mathbf{H}_T$  moins bien conditionnée) et les vitesses de convergence sont plus faibles. Le contraste et l'erreur de reproduction sont respectivement présentés en figures 7.10 et 7.11. Les courbes de l'al-

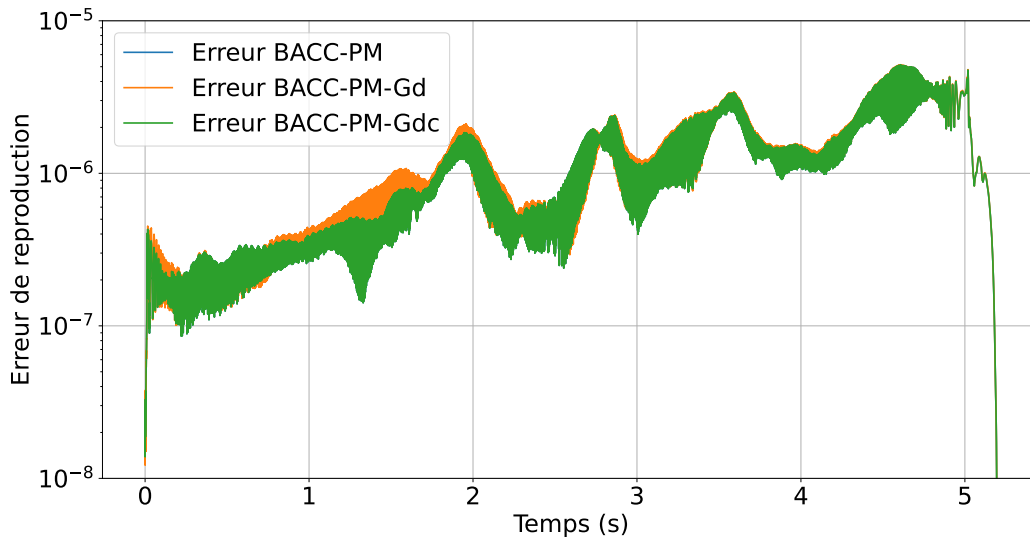


FIGURE 7.5 – Erreur de reproduction au cours du temps des méthodes BACC-PM, BACC-PM-Gd et BACC-PM-Gdc après 2000 itérations. Les courbes BACC-PM et BACC-PM-Gdc sont superposées.

gorithme BACC-PM et BACC-PM-Gdc sont à nouveau confondues, confirmant une fin de convergence de l'algorithme BACC-PM-Gdc à 2000 itérations (voire même dès 1000 itérations). Quant à l'algorithme BACC-PM-Gd, bien qu'ayant une erreur relative de 65 %, l'erreur de reproduction est du même ordre de grandeur que les deux autres algorithmes. Le contraste est moins bon de 5 dB jusqu'à 400 Hz mais reste supérieur à 30 dB. L'effort du réseau est affiché table 7.3, et à nouveau l'algorithme BACC-PM-Gd affiche un effort plus bas *a priori* à cause d'une *dark zone* incorrectement créée. De plus, au vu de la courbe de convergence de l'algorithme BACC-PM-Gd, la géométrie en mesure n'est pas optimale pour la création de *sound zones* (le système étant mal conditionné, i.e. la vitesse de convergence est lente). Il est donc normal de trouver des efforts du réseau plus importants qu'en simulation, cela revient à dire qu'il faut plus d'énergie pour créer les deux zones.

	BACC-PM	BACC-PM-Gd	BACC-PM-Gdc
EFF (dB)	11,4	6,6	11,4

TABLE 7.3 – Effort du réseau des méthodes BACC-PM, BACC-PM-Gd et BACC-PM-Gdc après 2000 itérations obtenu à partir des réponses impulsionnelles mesurées.

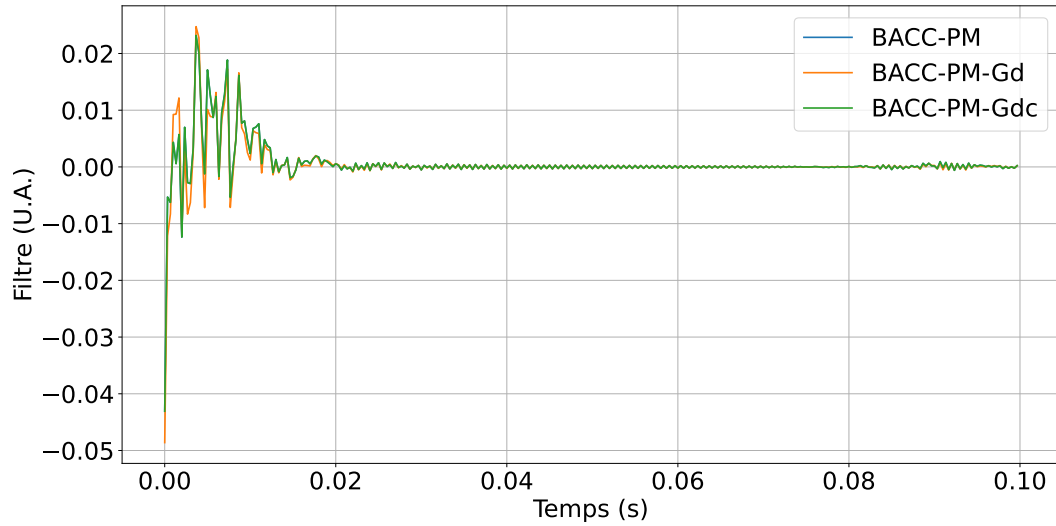


FIGURE 7.6 – Filtre appliqué au haut-parleur  $l = 0$  pour les méthodes BACC-PM, BACC-PM-Gd et BACC-PM-Gdc après 2000 itérations. Les courbes BACC-PM et BACC-PM-Gdc sont superposées.

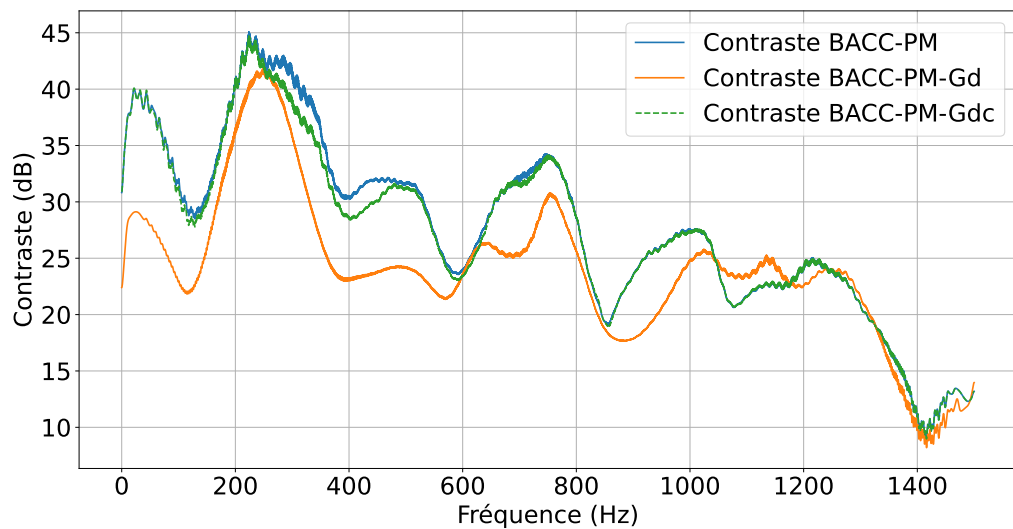


FIGURE 7.7 – Contraste en fonction de la fréquence des méthodes BACC-PM, BACC-PM-Gd et BACC-PM-Gdc après 100 itérations.



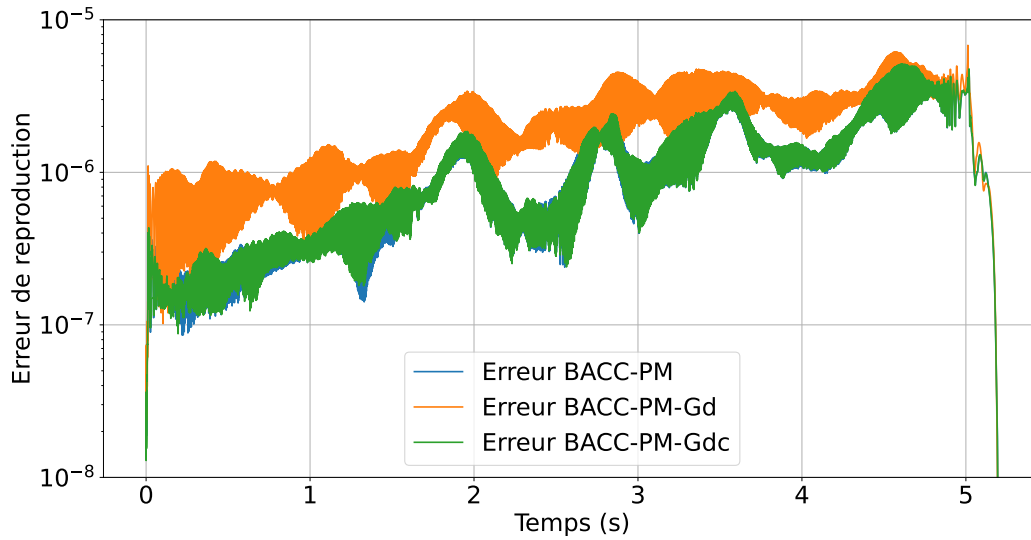


FIGURE 7.8 – Erreur de reproduction au cours du temps des méthodes BACC-PM, BACC-PM-Gd et BACC-PM-Gdc après 100 itérations. Les courbes BACC-PM et BACC-PM-Gdc sont superposées.

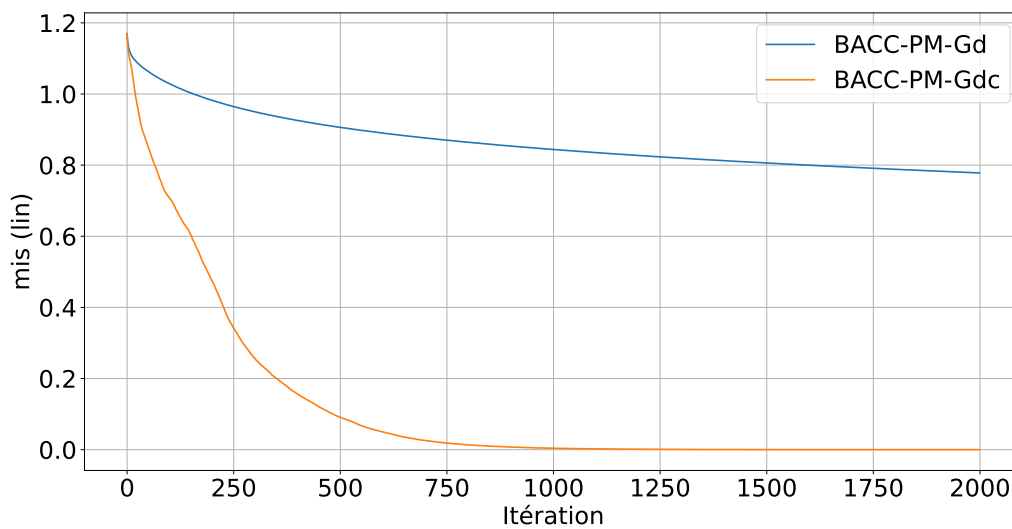


FIGURE 7.9 – Erreur de convergence BACC-PM-Gd et BACC-PM-Gdc obtenue à partir des réponses impulsionnelles mesurées.

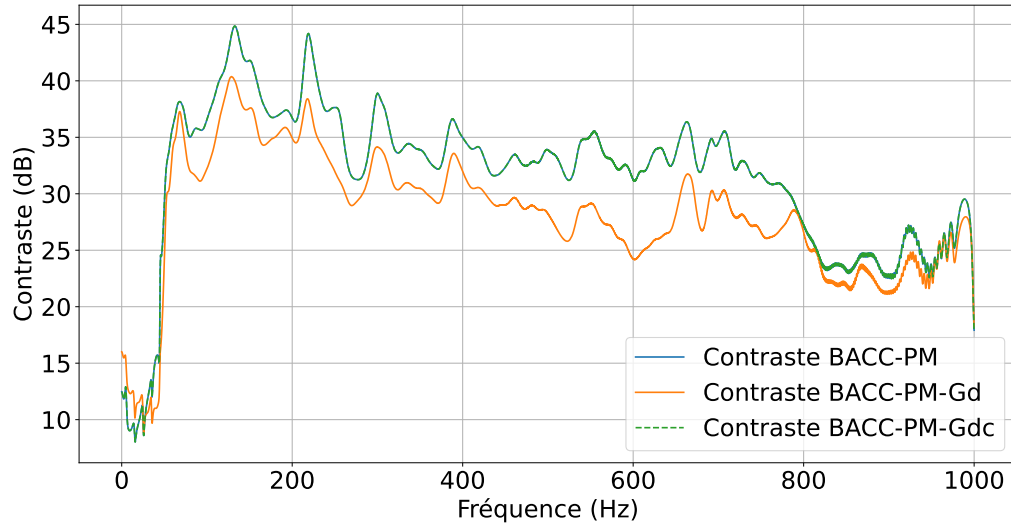


FIGURE 7.10 – Contraste en fonction de la fréquence des méthodes BACC-PM, BACC-PM-Gd et BACC-PM-Gdc après 2000 itérations en mesure. Les courbes BACC-PM et BACC-PM-Gdc sont superposées.

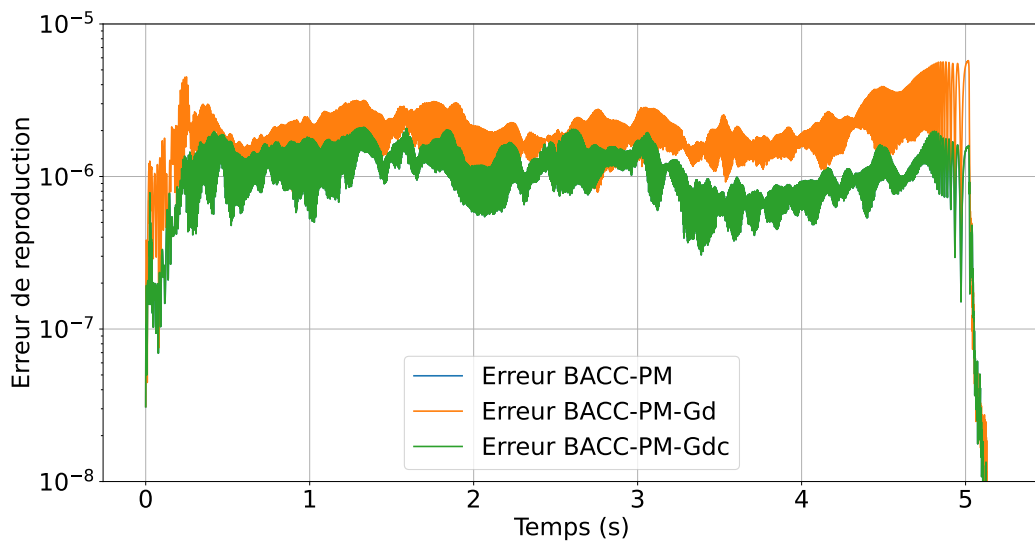


FIGURE 7.11 – Erreur de reproduction au cours du temps des méthodes BACC-PM, BACC-PM-Gd et BACC-PM-Gdc après 2000 itérations obtenue à partir des réponses impulsionnelles mesurées. Les courbes BACC-PM et BACC-PM-Gdc sont superposées.

## 7.3 Conclusion

L’algorithme BACC-PM a été modifié afin d’être résolu par descente de gradient (simple ou conjugué). Cette méthode d’optimisation récursive permet de mettre à jour les filtres de *sound zones* au cours du temps, ouvrant alors la voie aux cas variant dans le temps.

Une remarque peut être faite sur les filtres de *sound zones*, puisque les méthodes récursives permettent d’étudier le contraste pour des filtres éloignés de la solution visée. Sur des temps très court de convergence, les filtres estimés en cours sont éloignés des filtres à trouver, cependant les métriques propres aux *sound zones* montrent des performances intéressantes. Donc, malgré des temps de convergences relativement longs, les algorithmes peuvent être satisfaisants pour l’utilisateur très rapidement.

La descente de gradient conjugué présente un avantage certain sur la vitesse de convergence par rapport à la descente simple. Cependant, il reste à vérifier que ce type d’approche fonctionne également dans le cas variant dans le temps, ce qui fait l’objet du prochain chapitre.

# ZONES EN MOUVEMENT ET RÉPONSES IMPULSIONNELLES CONNUES

---

Dans ce chapitre, nous nous plaçons dans un cas variant dans le temps, en supposant par ailleurs que toutes les réponses impulsionnelles sont connues à tout instant.

Cela peut correspondre au cas d'un utilisateur qui se déplace d'une position à une autre, mais reste statique un long moment dans les deux positions. Par exemple, un utilisateur passe d'une chaise à un canapé. Le système peut donc être calibré aux deux endroits, et nous acceptons une baisse des performances pendant le déplacement.

Dans le chapitre précédent, l'algorithme BACC-PM a été modifié pour pouvoir résoudre le problème au cours du temps. Cette faculté de mettre à jour les filtres ouvre la voie à un cas variant dans le temps. En effet, dans ce cas les réponses impulsionnelles changent, donc il suffit de mettre à jour la matrice  $\mathbf{H}_T$  puis de mettre à jour les filtres.

## 8.1 Inversion de zones

La première simulation permet de vérifier que l'algorithme peut retrouver les filtres dans le cas d'un changement soudain et important. À  $t = 0$  s, les filtres  $w^{(l)}(n)$  sont initialisés par  $w^{(l)}(n) = \delta(n)$ ,  $\forall l$  et toutes les réponses impulsionnelles sont connues à tout instant.

L'équation de récurrence de mise à jour des filtres (7.9) est modifiée par :

$$\mathbf{w}_{k+1} = \mathbf{w}_k - \frac{1}{2} \frac{\mathbf{r}_k^\top \mathbf{r}_k}{\mathbf{r}_k^\top \mathbf{H}_T^\top(n) \mathbf{r}_k} \left( \mathbf{H}_T(n) \mathbf{w}_k - (1 - \beta) \mathbf{H}_B^\top(n) \mathbf{p}_t(n) \right), \quad (8.1)$$

avec

$$\mathbf{r}_k = -2\mathbf{H}_T(n) \mathbf{w}_k + 2(1 - \beta) \mathbf{H}_B^\top(n) \mathbf{p}_t(n). \quad (8.2)$$

L'indice  $k$  représente l'instant de mise à jour et l'indice  $n$  représente le temps. En effet, les réponses impulsionnelles ne changent qu'une seule fois au cours de l'expérience. Il est donc nécessaire de ne changer les matrices  $\mathbf{H}_T$ ,  $\mathbf{H}_B$  et le vecteur  $\mathbf{p}_t$  qu'une seule fois.

La géométrie au début de la simulation est présentée figure 8.1. La pression cible est créée en considérant le champ de pression naturel que créerait uniquement le HP 2 s'il émettait le signal  $x(n)$  sans filtrage spécifique. Chaque zone est discrétisée par une antenne carrée de neuf microphones espacés de 10 cm. La *bright zone* est à gauche au début de l'expérience et la *dark zone* à droite. À la moitié de l'expérience ( $t = 0,5$  s), les zones sont interverties. La salle simulée est une salle semi-anéchoïque, les haut-parleurs et les microphones étant positionnés à 1 m du sol. Les haut-parleurs sont modélisés par des monopoles acoustiques.

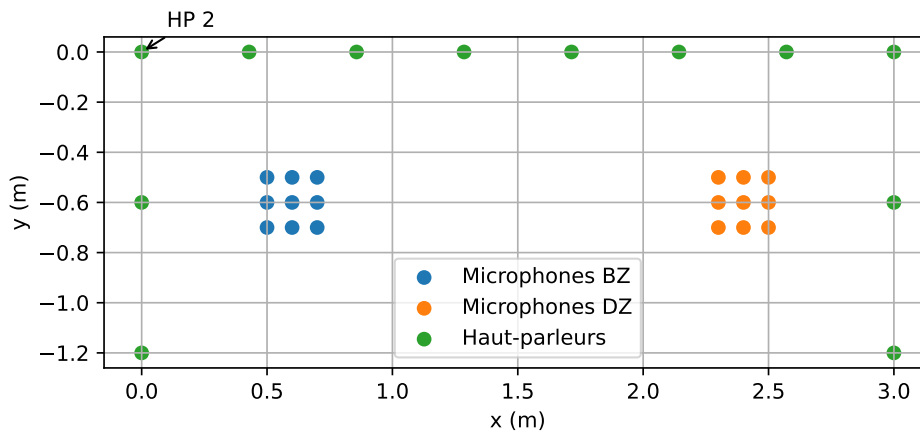


FIGURE 8.1 – Géométrie de la simulation d'inversion de zones à  $t = 0$  s. Le HP 2 permet de générer la pression cible au départ, puis le HP 9 génèrera la pression cible après l'inversion des zones.

La fréquence d'échantillonnage est de 3 kHz et les réponses impulsionnelles tout comme les filtres de *sound zones* contiennent 300 coefficients (100 ms). Le temps d'intégration

pour les pressions RMS est de 0,1 s. La simulation dure 1 s au total et les zones sont interverties à  $t = 0,5$  s ( $n = 1500$ ). Le HP 9 génèrera la pression cible après l'inversion des zones. Le contraste étant calculé comme le rapport des pressions RMS (équation (7.32)), il est pleine bande entre 0 Hz et la fréquence de Nyquist (1500 Hz ici).

Sur la figure 8.2, l'évolution du contraste moyen au cours du temps est présentée. Avant 200 ms, le contraste augmente, car les signaux doivent tout d'abord se propager dans les filtres puis dans la pièce. À partir de 100 ms, le contraste moyen s'établit autour de 22 dB. Sur la figure 8.3a, nous pouvons observer que le contraste en fréquence est correct au-delà de 800 Hz, ce très rapidement (150 ms). Ce sont les basses fréquences qui nécessitent plus de temps de convergence avant de retrouver les mêmes performances que l'inversion directe (voir figure 8.3b). À partir de 400 ms, l'inversion directe et la descente de gradient offrent les mêmes performances. L'erreur de reproduction (figure 8.2) décroît très rapidement pour s'établir autour de  $1 \times 10^{-4}$ , ce qui est comparable aux erreurs disponibles dans la littérature [7], [9], [16].

À  $t = 0,5$  s, lors de l'inversion des zones, les filtres de *sound zones* sont faux puisqu'ils correspondent au cas avant changement. Cependant, sur la courbe d'erreur de reproduction, nous remarquons qu'il suffit de 50 ms pour retrouver une erreur de  $1 \times 10^{-4}$ . Concernant le contraste en fréquence, nous pouvons remarquer sur la figure 8.3c le même comportement qu'au début : les hautes fréquences retrouvent plus rapidement le bon contraste que les basses fréquences. Enfin, 400 ms après le changement (figure 8.3d), l'algorithme par descente de gradient offre les mêmes performances que l'inversion directe.

Sur la figure 8.4 est présenté le filtre du haut-parleur 2 à quatre instants de la simulation. Pour la première partie de la simulation, avant 0,5 s, le haut-parleur 2 permet de générer la pression cible de la *bright zone* : ainsi le filtre doit se rapprocher d'un dirac. C'est bien ce que l'on observe pour les temps de 0,15 s et 0,4 s. Après l'inversion, à 0,55 s, le filtre est éloigné de la solution inversion directe, d'où les performances moindres présentées au-dessus. Par contre, à 0,93 s les deux filtres (BACC-PM et BACC-PM-Gd) sont proches, d'où les performances équivalentes.

Concernant l'algorithme BACC-PM-Gdc, le contraste au cours du temps et l'erreur de reproduction au cours du temps sont affichés figure 8.5. Par rapport à la descente de gradient classique, l'algorithme est un peu plus rapide sur la convergence de l'erreur de

reproduction, et le contraste s'établit rapidement autour de 23 dB. Par contre, en fréquence (figure 8.6), les performances sont identiques à la solution par inversion directe à tout instant, ce qui est confirmé par les filtres trouvés (figure 8.7), puisqu'ils sont identiques à chaque instant à ceux de l'inversion directe.

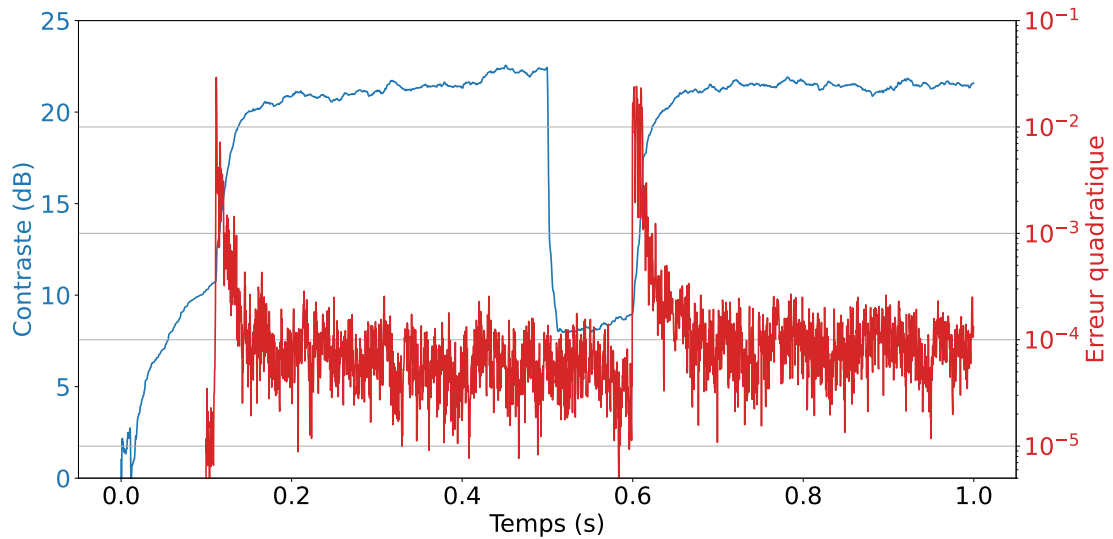


FIGURE 8.2 – Contraste et erreur de reproduction au cours du temps pour l'algorithme BACC-PM-Gd. Les zones sont interverties à  $t = 0,5$  s.

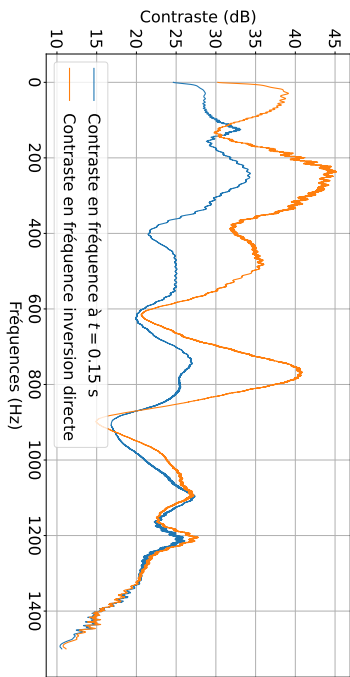
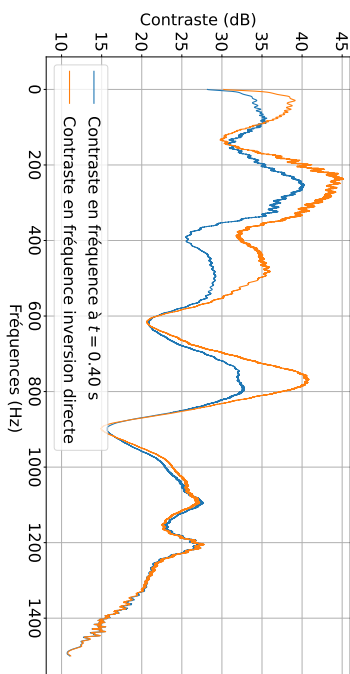
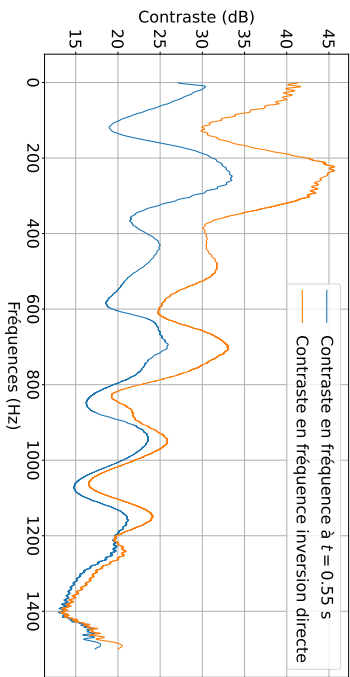
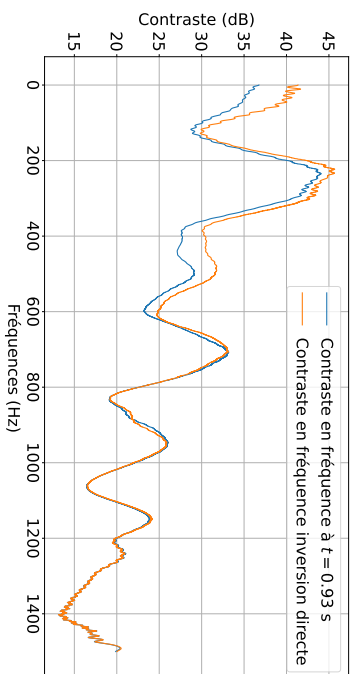
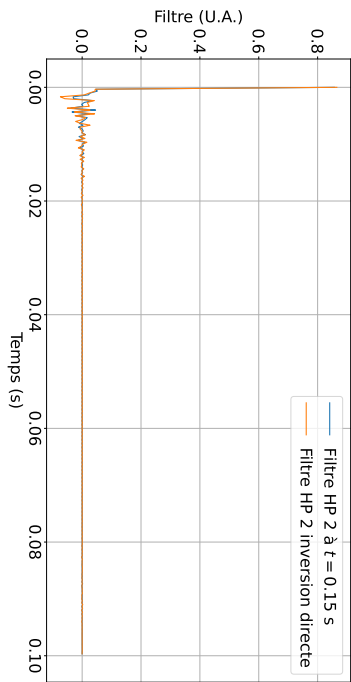
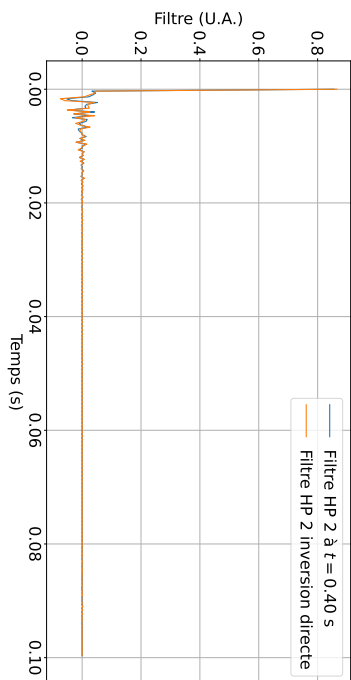
(a)  $t = 0,15$  s(b)  $t = 0,4$  s(c)  $t = 0,55$  s(d)  $t = 0,93$  s

FIGURE 8.3 – Contraste en fonction de la fréquence à différents instants pour l'algorithme BACC-PM-Gd.

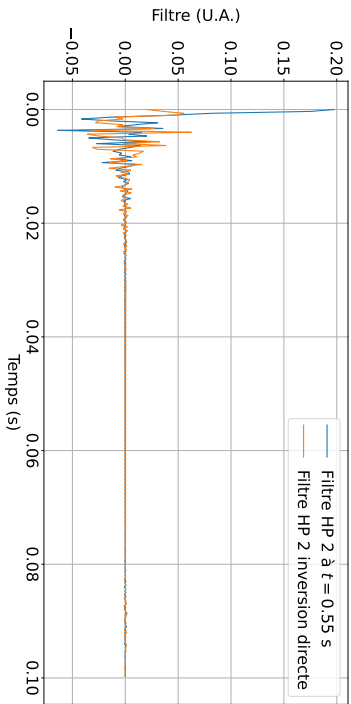




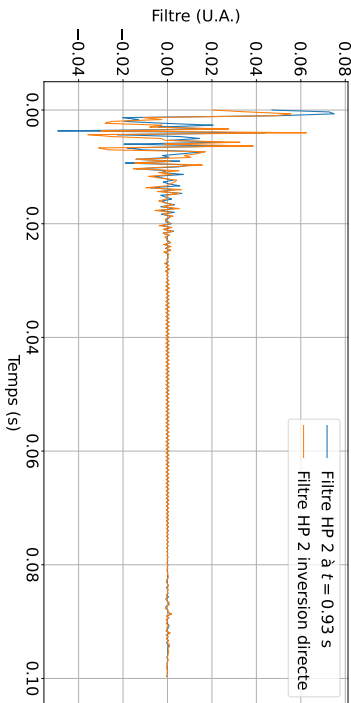
(a)  $t = 0,15$  s



(b)  $t = 0,4$  s



(c)  $t = 0,55$  s



(d)  $t = 0,93$  s

FIGURE 8.4 – Filtre du haut-parleur 2 de la géométrie 8.1 à différents instants pour l'algorithme BAOC-PM-Gd.

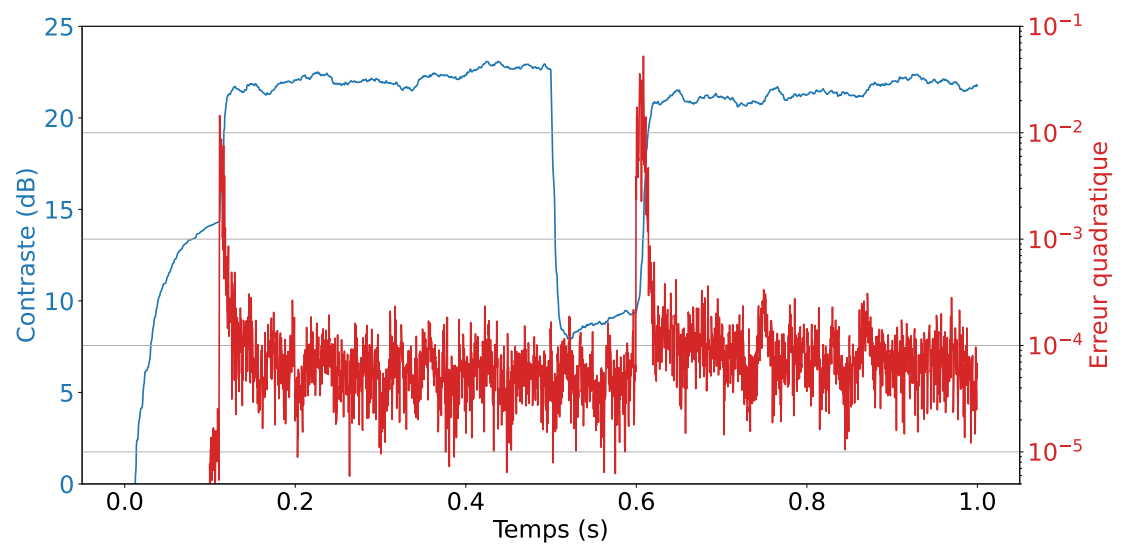
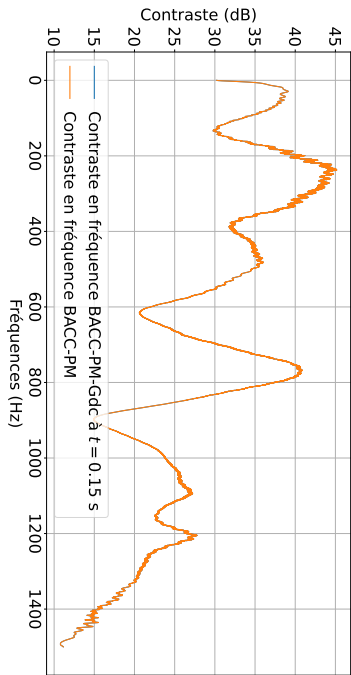
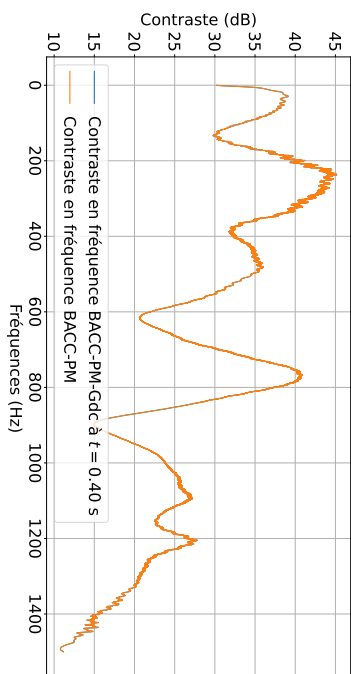


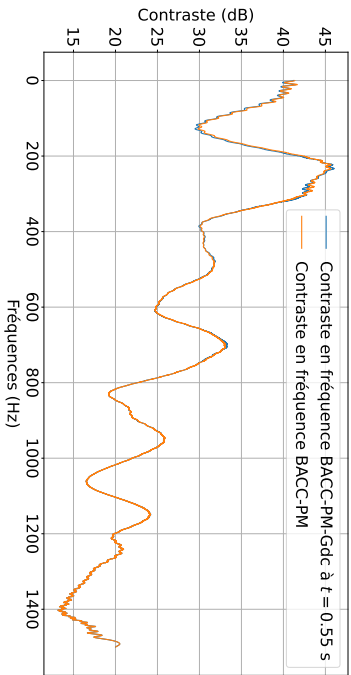
FIGURE 8.5 – Contraste et erreur de reproduction au cours du temps pour l’algorithme BACC-PM-Gdc. Les zones sont interverties à  $t = 0,5$  s.



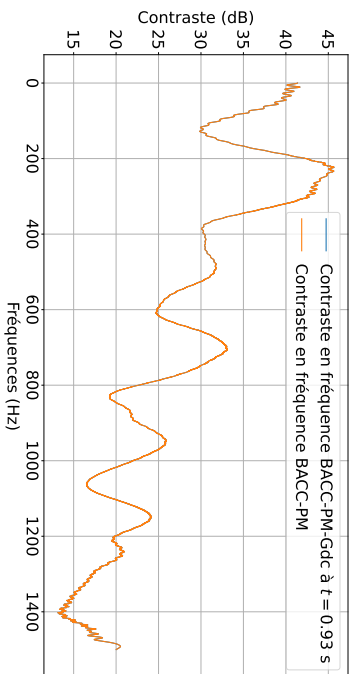
(a)  $t = 0,15$  s



(b)  $t = 0,4$  s



(c)  $t = 0,55$  s



(d)  $t = 0,93$  s

FIGURE 8.6 – Contraste en fonction de la fréquence à différents instants pour l'algorithme BACC-PM-Gdc.

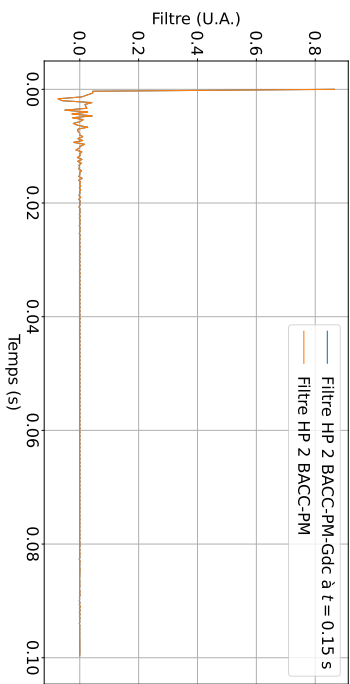
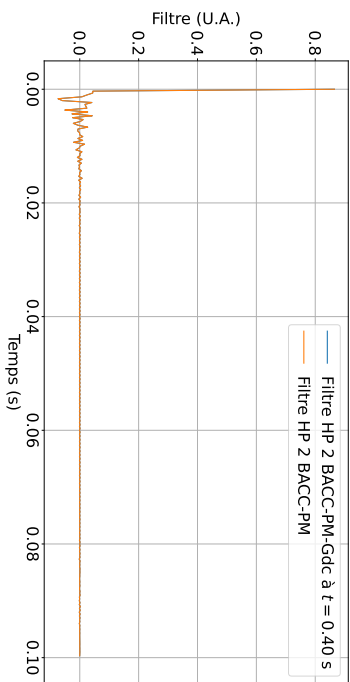
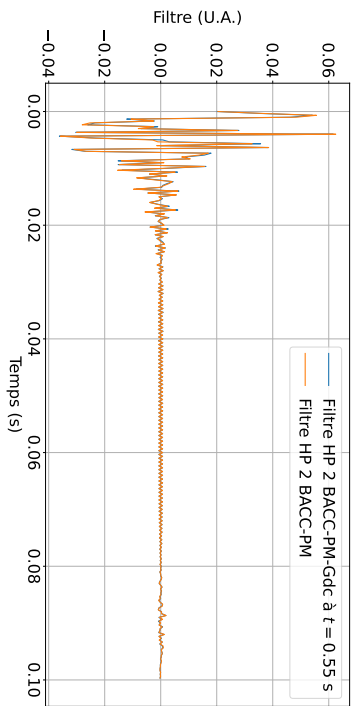
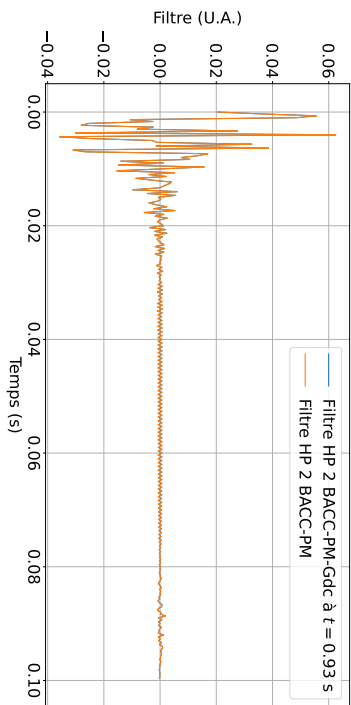
(a)  $t = 0,15$  s(b)  $t = 0,4$  s(c)  $t = 0,55$  s(d)  $t = 0,93$  s

FIGURE 8.7 – Filtre du haut-parleur 2 de la géométrie 8.1 à différents instants pour l'algorithme BACC-PM-Gdc.

## 8.2 Déplacement continu

Dans cette section, nous nous intéressons à un utilisateur mobile qui se déplace dans la pièce. Ainsi, la *bright zone* se déplace à  $1 \text{ m s}^{-1}$ , alors que la *dark zone* reste fixe<sup>1</sup>. Trois simulations sont présentées :

- la première ne suppose aucun *a priori* permettant de générer des filtres au départ qui sont alors initialisés par des diracs ; cela permet de vérifier que l'algorithme peut trouver une solution à partir d'une condition initiale arbitraire,
- la deuxième est initialisée au départ par inversion directe, ainsi à la première itération les filtres sont initialisés par la solution en inversion directe ; cela permet de vérifier que l'algorithme garde la bonne solution au cours du temps,
- enfin, la troisième simulation dégrade la connaissance des réponses impulsionnelles puisqu'elles sont supposées connues uniquement tous les 10 cm ; cela permet de vérifier que l'algorithme est robuste à une modélisation partielle du système.

La géométrie utilisée pour les trois simulations est identique, présentée figure 8.8.

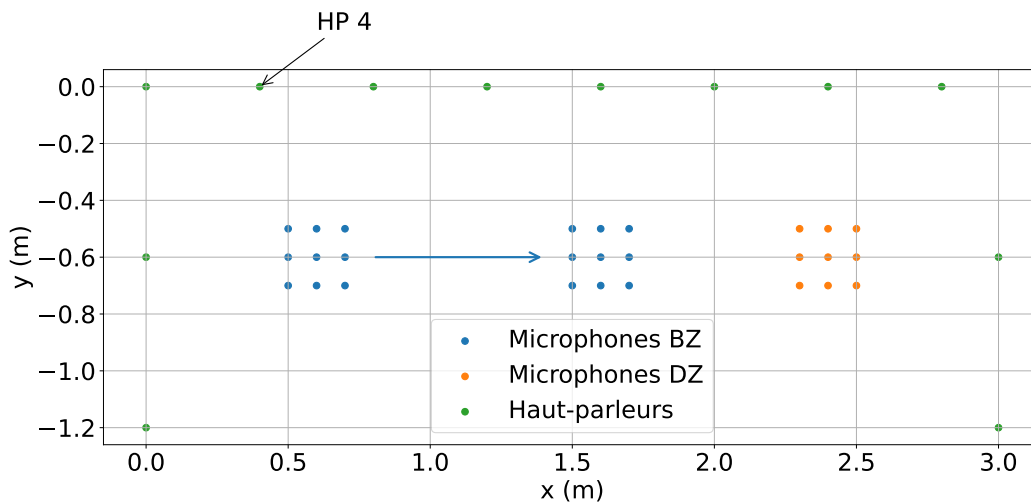


FIGURE 8.8 – Géométrie de la simulation lorsque la *bright zone* avance vers la droite. Le HP 4 permet de générer la pression cible.

1. Cette valeur est choisie proche de la valeur préférée de  $1,42 \text{ m s}^{-1}$  pour la marche [99].

**Note sur l'algorithme BACC-PM-Gdc :** La base de gradients conjugués est  $\mathbf{H}_T$ -orthogonale uniquement si la matrice  $\mathbf{H}_T$  ne varie pas au cours des itérations (et donc au cours du temps), ce qui n'est plus le cas ici. Dans la section précédente, la base a été réinitialisée au changement, or ici le changement a lieu à chaque instant  $n$ . En réinitialisant la base à chaque instant, alors, à l'itération courante, cela revient à effectuer une descente de gradient classique. Donc en réinitialisant la base à chaque instant, l'algorithme BACC-PM-Gdc devient un algorithme BACC-PM-Gd. Ainsi, seul ce dernier algorithme est utilisé ici.

### 8.2.1 Condition initiale nulle

Dans la première simulation, la *bright zone* avance à  $1 \text{ m s}^{-1}$  vers la droite. Le haut-parleur qui génère la pression cible est le numéro 4 (voir figure 8.8). Les réponses impulsionnelles sont connues à tout instant. Les filtres sont initialisés par une impulsion unitaire (dirac) au départ. La figure 8.9 présente le contraste au cours du temps et l'erreur au cours du temps. Il est à noter que le temps d'intégration des pressions RMS est de 0,1 s, ce qui explique le départ tronqué des courbes. Le contraste est à son maximum au départ, autour de 17 dB et l'erreur de reproduction s'établit dès 0,2 s à  $1 \times 10^{-4}$ . L'erreur de reproduction reste à sa valeur tout au long de la simulation et est identique au cas statique. Le mouvement ne pénalise donc pas cette erreur. Par contre, le contraste diminue régulièrement pour descendre en dessous de 15 dB à la fin du mouvement. Ceci est principalement dû au haut-parleur qui génère la pression cible qui devient éloigné de la zone à reproduire. De plus, la *dark zone* devient plus proche de la *bright zone* dans la configuration de la fin de la simulation.

### 8.2.2 Condition initiale fournie par BACC-PM

Les algorithmes adaptatifs présentent deux régimes de fonctionnement : un régime transitoire au départ pour trouver la solution à partir des conditions initiales (qui peuvent être très éloignées de la solution) et un régime permanent où l'algorithme doit rester le plus proche de la solution [25]. Selon que le système est invariant ou variant dans le temps, le régime permanent doit permettre de conserver la solution ou se réadapter aux petits changements respectivement. La simulation proposée ici concerne uniquement le régime permanent en initialisant les filtres de *sound zones* avec la solution trouvée par l'algorithme BACC-PM. Ainsi, dès le départ, les filtres sont justes, et l'algorithme BACC-

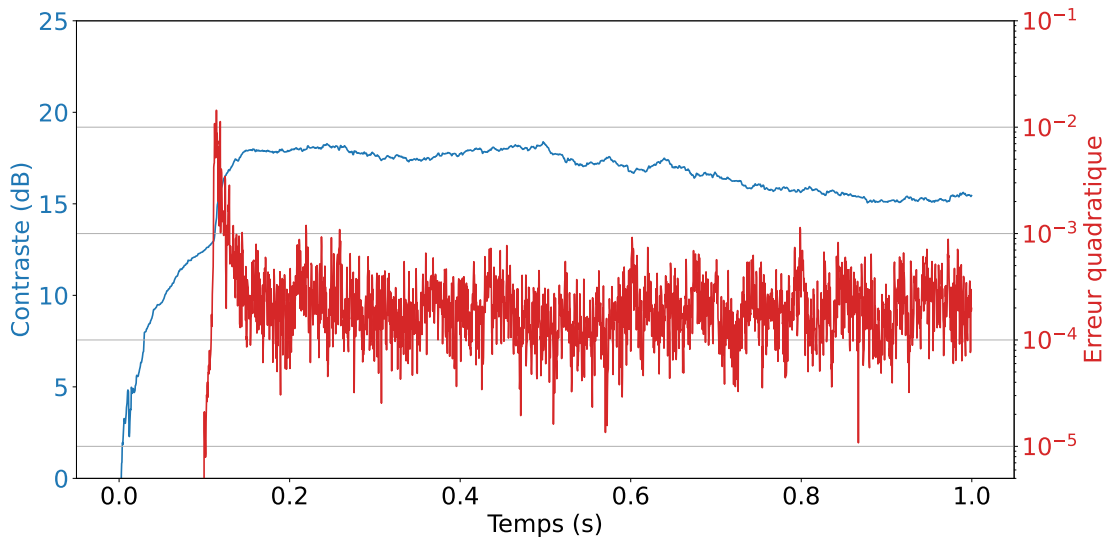


FIGURE 8.9 – Contraste moyen et erreur moyenne au cours du temps lorsque la *bright zone* avance à  $1 \text{ m s}^{-1}$  vers la droite. A  $t = 0 \text{ s}$  les filtres sont initialisés par des diracs.

PM-Gd devra uniquement adapter ces filtres en fonction du mouvement. La *bright zone* avance toujours à  $1 \text{ m s}^{-1}$  vers la droite. Le haut-parleur qui effectue la pression cible est le numéro 4 (le plus proche en moyenne de toutes les positions). Les réponses impulsionnelles sont connues à tout instant. Les filtres sont initialisés par les filtres de la position initiale trouvés par inversion directe.

La figure 8.10 présente le contraste au cours du temps et l'erreur au cours du temps. Par rapport à la simulation précédente, l'erreur au départ est immédiatement de  $1 \times 10^{-4}$  et le contraste présente un pic à 25 dB (contraste moyen maximal obtenu dans le cas statique). Très rapidement, le contraste chute à 17 dB, dès que la zone commence à bouger, et nous retrouvons alors une courbe très proche de la simulation précédente. L'erreur de reproduction reste autour de  $1 \times 10^{-4}$  tout le long de la simulation. Ainsi, la connaissance *a priori* des filtres de *sound zones* à  $t = 0 \text{ s}$  permet d'améliorer très légèrement les performances de l'algorithme au départ. Les performances de *tracking* de trajectoire restent limitées sur cet algorithme. Cependant, nous avons vu dans la section 8.1 que lorsque l'utilisateur s'arrête, les performances redeviennent rapidement acceptables. Dans le cas où un utilisateur passe d'une position A à une position B, fixes, il pourra donc bénéficier des performances des méthodes de *sound zones* dans les deux positions. Par contre, lors du déplacement, le contraste chutera.

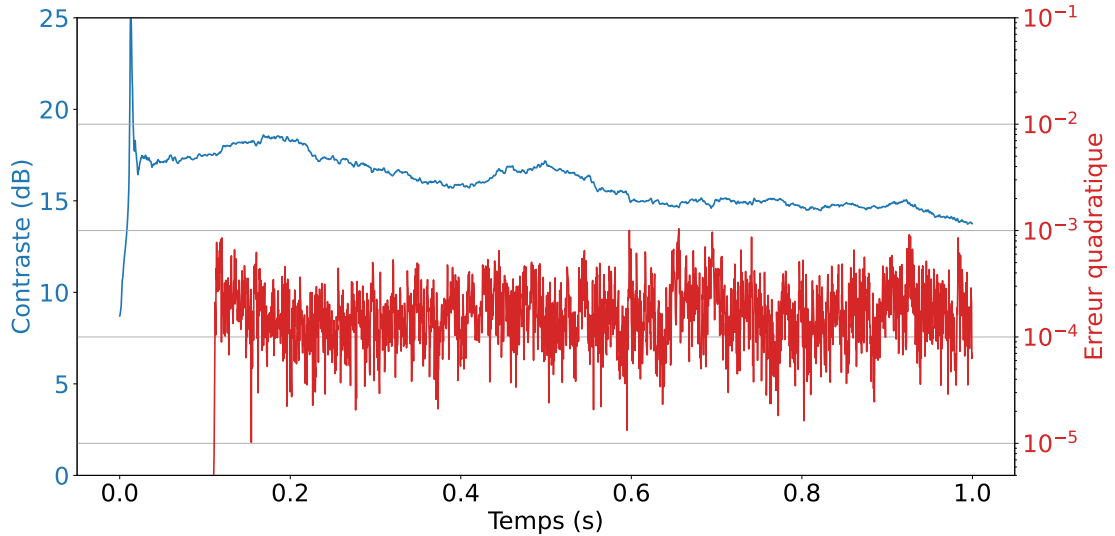


FIGURE 8.10 – Contraste moyen au cours du temps lorsque la *bright zone* avance à  $1 \text{ m s}^{-1}$  vers la droite. À  $t = 0 \text{ s}$  les filtres sont initialisés par les filtres générés par inversion directe.

Il est également à noter que la vitesse a une faible incidence sur la capacité de suivi de l'algorithme. Par exemple, sur la figure 8.11 les courbes de contraste et erreur de reproduction sont tracées au cours du temps pour une vitesse deux fois moindre ( $0,5 \text{ m s}^{-1}$ ). Les résultats obtenus sont très proches de ceux présentés en figure 8.10, à  $1 \text{ m s}^{-1}$ .

### 8.2.3 Réponses impulsionnelles connues tous les 10 cm

Jusqu'à présent, les réponses impulsionnelles sont supposées connues à chaque instant et pour toutes les positions. En pratique, il n'est pas possible de réaliser des mesures aussi fines. L'algorithme testé ici suppose les réponses impulsionnelles connues uniquement tous les 10 cm. Entre deux positions, ce sont les réponses impulsionnelles précédentes qui sont utilisées et qui restent donc fixes sur les 10 cm suivants. La *bright zone* avance toujours à  $1 \text{ m s}^{-1}$  vers la droite. Le haut-parleur qui effectue la pression cible est le numéro 4 (le plus proche en moyenne de toutes les positions). Les réponses impulsionnelles sont connues à tout instant. Les filtres sont initialisés par des diracs. Les réponses impulsionnelles sont mises à jour tous les 10 cm pour le calcul des filtres. La figure 8.12 présente le contraste au cours du temps et l'erreur au cours du temps.



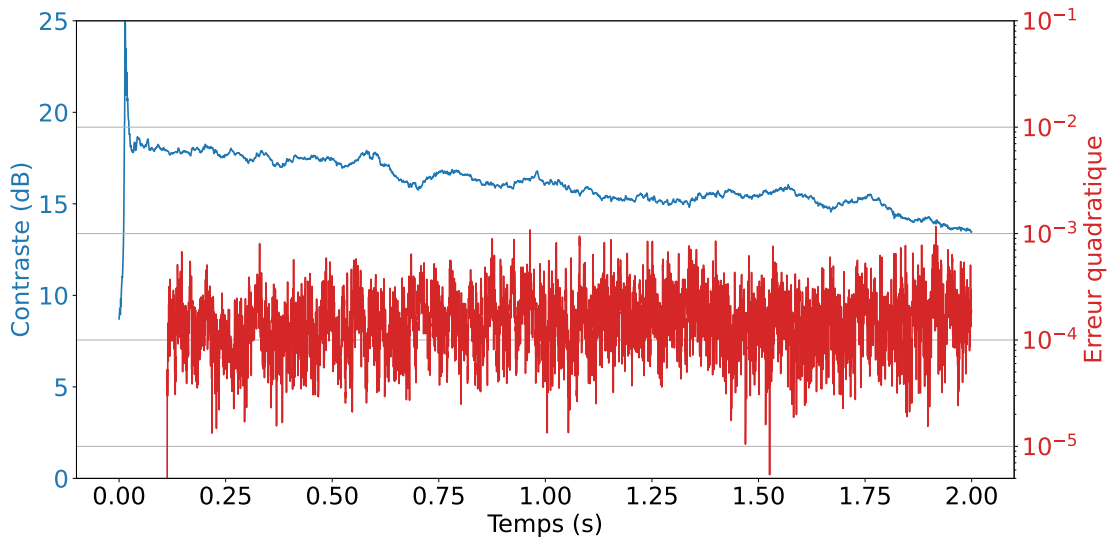


FIGURE 8.11 – Contraste moyen au cours du temps lorsque la *bright zone* avance à  $0,5 \text{ m s}^{-1}$  vers la droite. À  $t = 0 \text{ s}$  les filtres sont initialisés par les filtres générés par inversion directe.

Concernant le contraste, nous retrouvons le même comportement que lors des deux simulations précédentes : partant d'une valeur de 17 dB, le contraste diminue ensuite de manière régulière jusqu'à 15 dB à la fin du mouvement. La dégradation des connaissances des réponses impulsionnelles a donc, dans ce cas, très peu d'influence sur le contraste. L'erreur de reproduction représente quant à elle une oscillation entre  $1 \times 10^{-4}$  et  $1 \times 10^{-3}$ . Logiquement, l'erreur est plus grande lorsque les réponses impulsionnelles sont les plus éloignées de la position courante. Il est toutefois à noter que, lorsque les réponses impulsionnelles sont mises à jour et coïncident avec les réponses impulsionnelles de la position de calcul, l'erreur de reproduction retrouve très rapidement sa valeur de  $1 \times 10^{-4}$ .

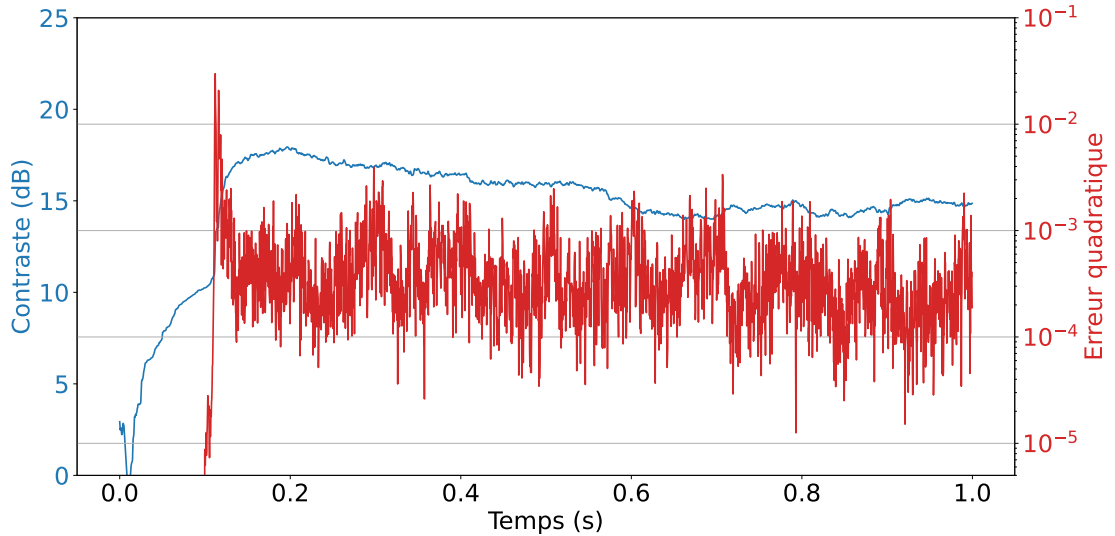


FIGURE 8.12 – Contraste moyen et erreur moyenne au cours du temps lorsque la *bright zone* avance à  $1 \text{ m s}^{-1}$  vers la droite. A  $t = 0$  les filtres sont initialisés par des diracs. Les RI sont mises à jour tous les 10 cm.

### 8.3 Conclusion

Dans cette section, l’algorithme BACC-PM-Gd a été testé dans deux configurations où les zones varient dans le temps : une inversion des zones et un mouvement continu de l’une des zones. L’algorithme BACC-PM-Gdc peut offrir des performances supérieures en terme de vitesse de convergence dans des cas où la matrice  $\mathbf{H}_T$  est constante sur une période, cependant dès qu’elle change à chaque instant alors la réinitialisation de la base de vecteurs orthogonaux revient alors à effectuer une descente de gradient classique.

Dans le cas quasi-statique, i.e. deux cas statiques juxtaposés, l’algorithme BACC-PM permet de trouver les filtres dans la première position, puis se réadapte à la seconde position en une centaine de millisecondes. Cependant, cette situation ne peut être réalisée en réalité, car l’utilisateur a besoin d’un certain temps pour passer de la première à la seconde position. Qu’il se déplace à  $0,5 \text{ m s}^{-1}$  ou  $1 \text{ m s}^{-1}$ , le contraste chute fortement (d’un facteur deux dans notre cas), bien que l’erreur de reproduction reste du même ordre de grandeur que dans le cas statique. L’utilisateur percevra une dégradation des performances durant le mouvement, mais une fois dans sa nouvelle position, fixe, les performances redeviendront identiques à celles du départ.

Dans tout ce chapitre, les réponses impulsionnelles sont supposées connues, à minima tous les 10 cm. Or cela nécessite de réaliser de nombreuses mesures en amont, ce qui peut s'avérer complexe à mettre en œuvre dans la pratique, ce d'autant plus que le mouvement des zones peut être quelconque. Dans la première partie de ce document, une méthode d'estimation de réponses impulsionnelles a été présentée s'appuyant sur la seule connaissance des signaux émis et mesurés. Cette méthode peut donc compléter l'algorithme détaillé dans ce chapitre afin, conjointement, d'estimer les réponses impulsionnelles et de résoudre le problème de *sound zones*. C'est l'objet du prochain chapitre qui esquisse une synthèse des deux parties de ce travail sous la forme de perspectives de travaux à venir.

# ZONES EN MOUVEMENT ET RÉPONSES IMPULSIONNELLES INCONNUES : PERSPECTIVES

---

Dans ce dernier chapitre à caractère prospectif, l'algorithme MISO-Autostep de la première partie (chapitre 3) est utilisé pour l'estimation des réponses impulsionnelles au cours du temps et, dans le même temps, l'algorithme BACC-PM-Gd (chapitre 7) permet de générer les filtres de *sound zones*.

Nous reprenons la même géométrie que celle du chapitre précédent (voir figure 8.1), avec  $L = 12$  haut-parleurs et  $M = 18$  microphones. Les réponses impulsionnelles sont dans un premier temps simulées dans une salle semi-anéchoïque selon la formule (7.29). La fréquence d'échantillonnage est  $F_s = 3$  kHz et le nombre de coefficients pour les réponses impulsionnelles et des filtres de *sound zones* est  $N = 300$ .

Pour rappel, l'équation générale du système est :

$$p^{(m)}(n) = \sum_{l=1}^L \left( h^{(m,l)} * y^{(l)} \right) (n), \quad (9.1)$$

avec

$$y^{(l)}(n) = \left( w^{(l)} * x \right) (n). \quad (9.2)$$

Comme il y a  $M$  microphones, il est nécessaire d'utiliser  $M$  algorithmes MISO-Autostep en parallèle (soit un par microphone). Chaque algorithme prend en entrée les  $L$  signaux  $y^{(l)}$  et le signal microphonique  $p^{(m)}$ . Il retourne alors une estimation des réponses impulsionnelles  $h^{(m,l)}(n), \forall l$ , valides à l'instant  $n$ .

Ces estimations de réponses impulsionnelles permettent alors de construire les matrices  $\mathbf{H}_T$  et  $\mathbf{H}_B$  ainsi que les vecteurs  $\mathbf{r}_k$  et  $\mathbf{p}_t$ . Puis, grâce à l'équation (7.9), les filtres de *sound zones* sont mis à jour. Deux configurations peuvent être analysées : celle en boucle ouverte et celle en boucle fermée. Dans la configuration « boucle ouverte », pour générer les zones, le premier algorithme testé est BACC-PM-Gd en ne prenant pas en compte le signal d'entrée ni la mesure de l'erreur réelle. Ce sont donc les équations de la section 7.1 qui sont utilisées.

**Note sur la corrélation :** Les signaux d'entrée des algorithmes MISO-Autostep étant tous construits à partir du signal  $x(n)$ , commun à tous les haut-parleurs, ils sont cette fois-ci corrélés entre eux, alors que l'algorithme MISO-Autostep fait l'hypothèse de décorrélation des différents signaux d'entrée. Cependant, dans l'article de HU *et al.* [19], les auteurs montrent que, tant que les filtres de *sound zones*  $w^{(l)}(n)$  sont non nuls et varient dans le temps (ce qui est notre cas), il est possible de déterminer les réponses impulsionnelles  $h^{m,l}(n)$ .

**Note sur l'initialisation :** Si les conditions initiales des filtres de *sound zones* étaient jusqu'à présent des diracs, cette initialisation pose problème pour les algorithmes MISO-Autostep, car ils risquent de converger vers la solution triviale de somme des RI. En effet, si tous les filtres sont identiques ( $w(n) = w^{(l)}(n), \forall l$ ), alors  $y^{(l)}(n) = y(n)$ , et par conséquent le signal de pression au microphone  $m$  s'écrit :

$$\begin{aligned} p^{(m)}(n) &= \sum_{l=1}^L \left( h^{(m,l)} * y^{(l)} \right) (n) \\ &= \sum_{l=1}^L \left( h^{(m,l)} * y \right) (n) \\ &= \left( \left( \sum_{l=1}^L h^{(m,l)} \right) * y \right) (n). \end{aligned} \tag{9.3}$$

La somme des réponses impulsionnelles est donc une solution du problème d'estimation ! Il est donc nécessaire de s'assurer que les filtres ne sont pas strictement égaux. Pour éviter cette situation, les filtres sont initialisés par des valeurs aléatoires  $w^{(l)} = \mathcal{N}(0, 1/L)$ .

## 9.1 Algorithmes MISO-Autostep et BACC-PM-Gd (boucle ouverte)

La première simulation correspond à une validation de la technique en statique. Les algorithmes MISO-Autostep sont initialisés avec les réponses impulsionnelles de la salle, comme si elles avaient été mesurées au début. Les méta-paramètres sont tous choisis tels que  $(\kappa, \gamma) = (1 \times 10^{-2}, 1 \times 10^{-4})$ . L'algorithme BACC-PM-Gd doit donc trouver les filtres de *sound zones* à partir des réponses impulsionnelles estimées, qui elles ne doivent pas changer puisque le système est statique.

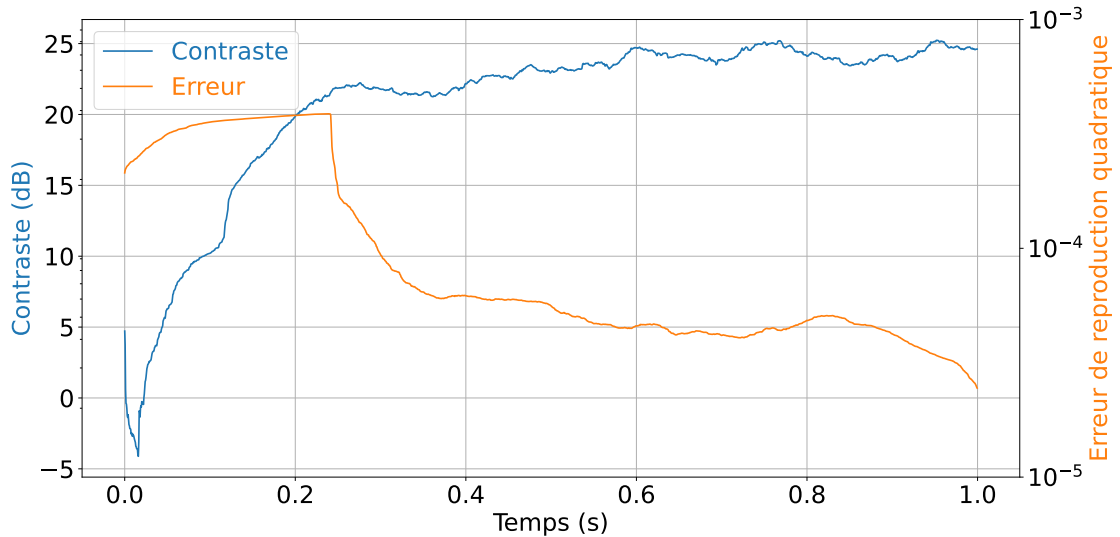


FIGURE 9.1 – Contraste et erreur de reproduction au cours du temps pour l'algorithme BACC-PM-Gd et  $M$  MISO-Autostep dans un cas statique.

Sur la figure 9.1 sont présentés le contraste et l'erreur de reproduction au cours du temps. Le contraste augmente jusqu'à 25 dB et l'erreur de reproduction diminue en dessous de  $1 \times 10^{-4}$  en 0,4s. Donc la génération des zones par les filtres obtenus avec l'algorithme BACC-PM-Gd n'a pas d'influence sur l'estimation des réponses impulsionnelles (en statique) par l'algorithme MISO-Autostep. Par contre, lorsqu'il y a inversion de zones (figure 9.2), les métriques contraste et erreur de reproduction sont dégradées. En effet, dans ce cas la *dark zone* est correctement créée avant l'inversion, puis à  $t = 0,2$ s (après l'inversion), l'algorithme MISO-Autostep ne parvient plus à estimer les réponses impulsionnelles des

microphones de l'ancienne *dark zone*, car le niveau est trop faible. Comme les estimations des réponses impulsionnelles ne changent pas (ou peu), l'algorithme BACC-PM-Gd ne parvient donc pas à mettre à jour les filtres de *sound zones* pour cette nouvelle géométrie, et le système reste bloqué dans la position précédente (avant inversion). Pour cette raison, le contraste devient négatif et l'erreur de reproduction passe de  $1 \times 10^{-3}$  à  $1 \times 10^{-2}$ .

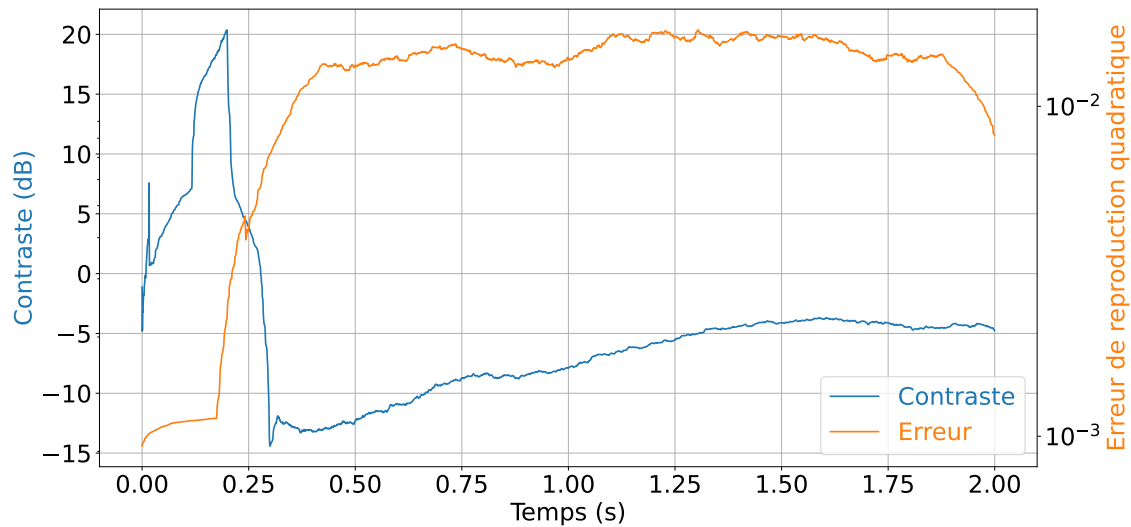


FIGURE 9.2 – Contraste et erreur de reproduction au cours du temps pour l'algorithme BACC-PM-Gd et  $M$  MISO-Autostep avec une inversion des zones à  $t = 0,2$ s.

Pour contourner ce problème, une possibilité est d'utiliser un algorithme en boucle fermée, où les estimations des réponses impulsionnelles ne sont pas directement utilisées pour la résolution du problème inverse, mais uniquement pour aider l'algorithme.

## 9.2 Algorithmes MISO-Autostep et BACC-PM-FxLMS (boucle fermée)

En boucle ouverte, l'algorithme n'a pas connaissance des pressions dans les différentes zones. Il s'appuie uniquement sur les estimations des réponses impulsionnelles données par les algorithmes MISO-Autostep. Or, ces estimations peuvent être de mauvaise qualité et les métriques dans les zones, de valeurs inférieures aux valeurs cibles attendues.

Cependant, la présence des microphones dans les deux zones permet d'accéder à une information sur les valeurs réelles des pressions et donc sur l'erreur réelle. Ces informations peuvent alors être utilisées dans une perspective « boucle fermée ». À noter que dans certaines applications, si la mise en place de microphones peut gêner l'utilisateur, il est cependant possible d'utiliser des antennes virtuelles pour reconstruire les pressions aux points de l'espace manquants [91].

La pression cible dans la *dark zone* étant nulle, et l'erreur de reproduction dans la *bright zone* étant définie comme la différence de la pression dans la *bright zone* et de la pression désirée  $\mathbf{p}_t$ , l'erreur pour tous les microphones est alors :

$$\mathbf{e}(n) = \begin{bmatrix} \mathbf{Q}_B(n) \\ \mathbf{Q}_D(n) \end{bmatrix} \mathbf{w} - \begin{bmatrix} \mathbf{p}_t(n) \\ \mathbf{0} \end{bmatrix} = \mathbf{Q}(n)\mathbf{w} - \bar{\mathbf{p}}_t(n). \quad (9.4)$$

Le Lagrangien associé à la minimisation de cette erreur quadratique, en imposant une contrainte sur l'énergie disponible du réseau, s'écrit alors :

$$J(\mathbf{w}) = \|\mathbf{e}(n)\|^2 + \lambda\|\mathbf{w}\|^2. \quad (9.5)$$

Le gradient de la fonction coût s'écrit donc sous la forme :

$$\begin{aligned} \nabla J &= 2\mathbf{Q}^\top(n)\mathbf{Q}(n)\mathbf{w} - 2\mathbf{Q}^\top(n)\mathbf{p}_t(n) + \lambda\mathbf{w} \\ &= 2\mathbf{Q}^\top(n) \left( \mathbf{Q}^\top(n)\mathbf{w} - \mathbf{p}_t(n) \right) + \lambda\mathbf{w} \\ &= 2\mathbf{Q}^\top(n)\mathbf{e}(n) + \lambda\mathbf{w}. \end{aligned} \quad (9.6)$$

Nous recherchons alors la valeur de  $\mathbf{w}$  par descente de gradient telle que :

$$\mathbf{w}_{k+1} = \mathbf{w}_k - \frac{1}{2}\theta\nabla J(\mathbf{w}), \quad (9.7)$$

avec  $\theta$  le paramètre de descente du gradient, soit, en reportant la valeur du gradient dans l'équation précédente :

$$\mathbf{w}_{k+1} = (1 - \lambda\theta)\mathbf{w}_k - \mathbf{Q}^\top(n)\mathbf{e}(n), \quad (9.8)$$

expression de la mise à jour d'un algorithme *Leaky FxLMS* [100] dans le cas où  $\lambda > 0$ , algorithme qui sera par la suite nommé BACC-PM-FxLMS.



Dans ce cas, c'est l'erreur mesurée qui est minimisée. Les estimations des réponses impulsionnelles permettent de construire la matrice  $\mathbf{Q}(n)$ , mais seule une estimation grossière est nécessaire pour assurer la convergence de l'algorithme [101].

Pour valider le fonctionnement de l'algorithme, une première simulation est effectuée dans le cas où les réponses impulsionnelles sont connues et les zones sont immobiles. La figure 9.3 présente le contraste et l'erreur de reproduction au cours du temps. Le paramètre de descente de gradient  $\theta$  est ici égal à  $2 \times 10^{-4}$ . Nous pouvons remarquer que la convergence est bien plus lente que dans le cas où le signal n'est pas pris en compte. L'erreur de reproduction converge vers une valeur de  $1 \times 10^{-4}$  au bout de 80 s. Le contraste quant à lui augmente très rapidement à 15 dB, en 5 s, puis ralentit fortement pour se rapprocher de 20 dB au bout de 100 s.

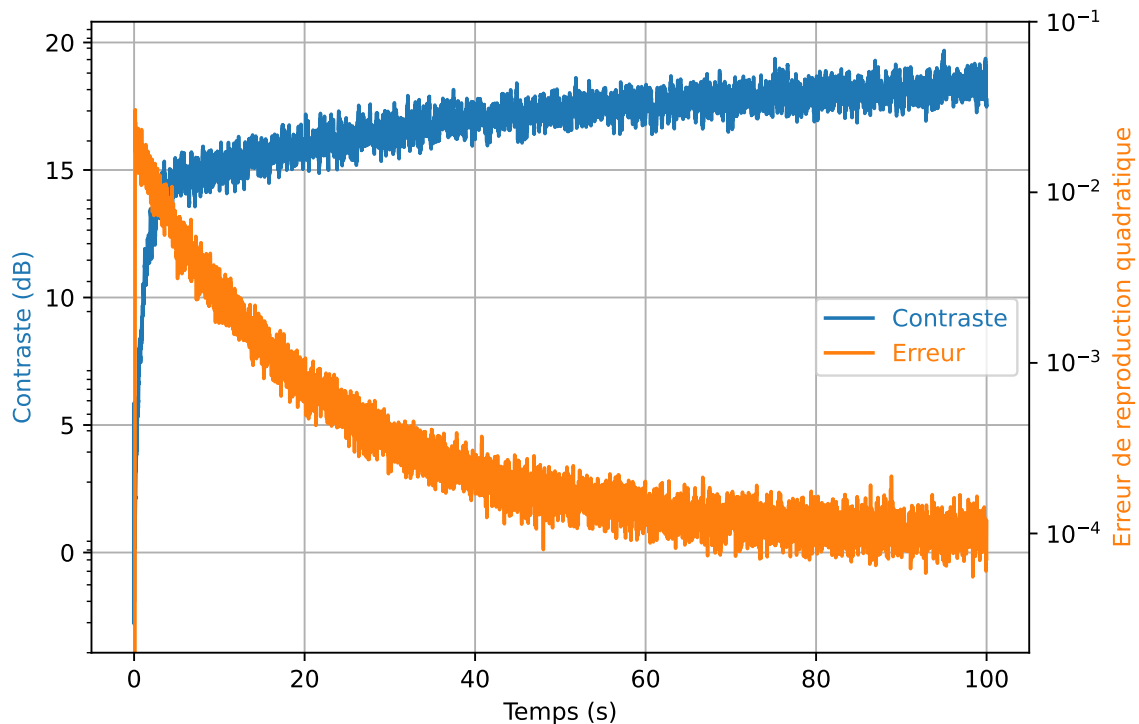


FIGURE 9.3 – Contraste et erreur de reproduction au cours du temps pour l'algorithme BACC-PM-FxLMS.

La différence de contraste par rapport à l'inversion directe s'explique par la courbe de contraste en fréquence proposée en figure 9.4. Nous remarquons que le contraste est de 10 dB inférieur à la valeur obtenue par inversion directe, et ce pour toutes les fréquences.

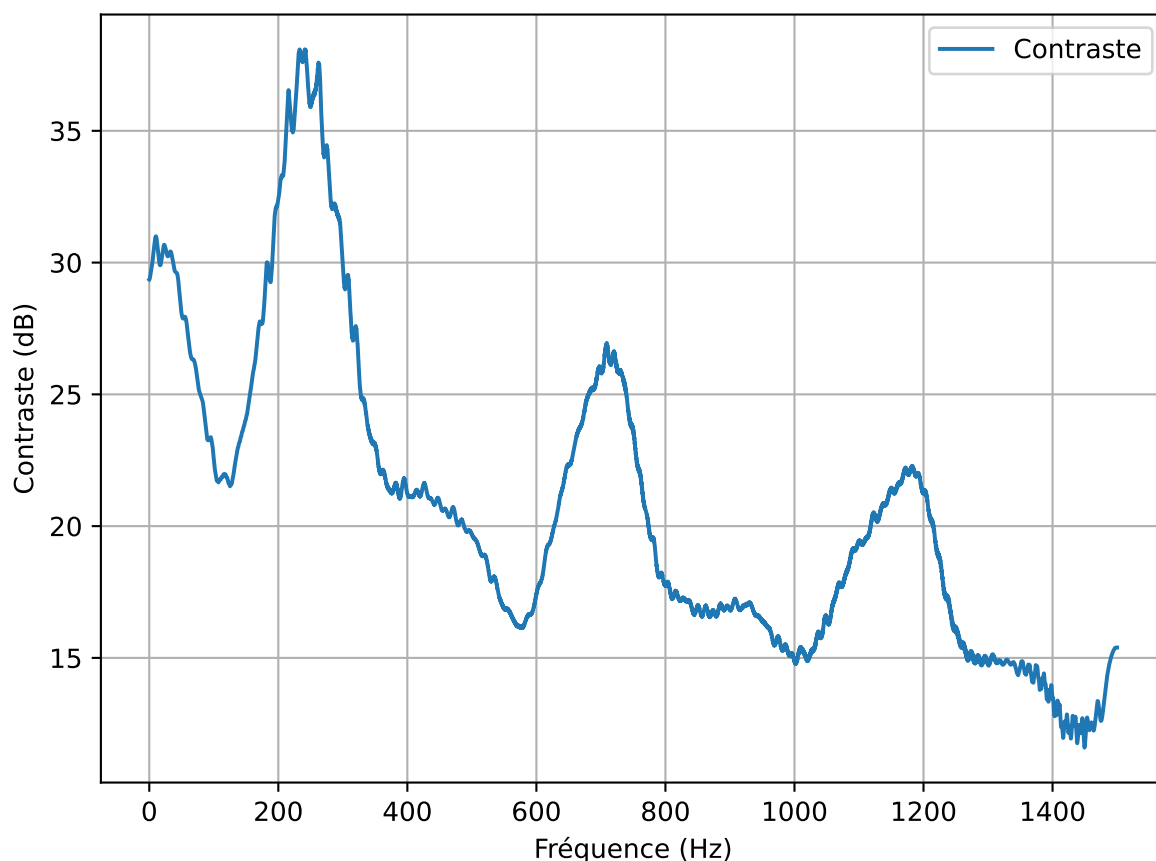


FIGURE 9.4 – Contraste en fonction de la fréquence pour l'algorithme BACC-PM-FxLMS à la fin de la simulation.

Maintenant que l'algorithme BACC-PM-FxLMS est fonctionnel, nous pouvons utiliser les algorithmes MISO-Autostep pour estimer les réponses impulsionnelles et construire la matrice  $\mathbf{Q}(n)$ . Il y a donc  $M$  algorithmes MISO-Autostep qui fonctionnent en parallèle (un par microphone). Ils prennent tous en entrée les signaux  $y^{(l)}(n)$ , puis chacun a en sortie le signal  $p^{(m)}(n)$  du microphone sur lequel il travaille. La principale différence avec le fonctionnement des algorithmes décrits dans la partie I est liée à la corrélation des signaux d'entrée. En effet, ils se basent tous sur le signal unique à diffuser  $x(n)$ .

Les filtres MISO-Autostep sont initialisés par des vecteurs nuls (y compris les réponses impulsionnelles  $h^{(m,l)}(n)$ ) comme dans le chapitre 3, les méta-paramètres sont tous choisis tels que  $(\kappa, \gamma) = (1 \times 10^{-2}, 1 \times 10^{-4})$ . Concernant l'algorithme BACC-PM-FxLMS, les paramètres associés à la partie *sound zones* sont choisis comme dans la section précédente :  $\beta = 0,5$  et  $\lambda = 1 \times 10^{-7}$ . Le pas de descente de gradient est choisi tel que  $\theta = 2 \times 10^{-6}$ .

Enfin, un bruit de fond blanc gaussien centré à 30 dB SPL aux signaux microphoniques est ajouté<sup>1</sup>.

Ayant  $M$  MISO-Autostep en parallèle, nous obtenons  $M$  courbes d'erreurs  $e^{(m)}(n)$ . Pour obtenir la courbe de convergence de la partie « estimation des réponses impulsionnelles », nous effectuons une moyenne à chaque instant sur les  $M$  microphones de l'erreur quadratique instantanée, puis la courbe est lissée sur  $N_w = 500$  points suivant :

$$\text{EQM}(n) = \frac{1}{N_w} \sum_{k=n-N_w}^n \frac{1}{M} \sum_{m=1}^M \|e^{(m)}(n)\|^2. \quad (9.9)$$

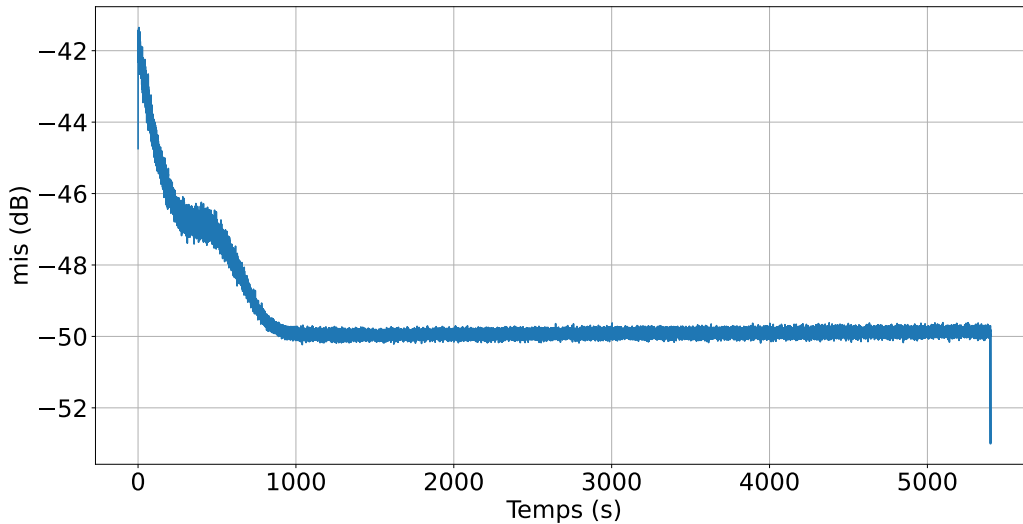


FIGURE 9.5 – Erreur quadratique moyenne des  $M$  MISO-Autostep.

---

1. Dans le cas où il n'y a pas de bruit, des simulations annexes montrent que la stabilité n'est pas assurée. En effet, lorsque l'algorithme de *sound zones* fonctionne bien, la pression dans la *dark zone* devient quasiment nulle, et les algorithmes MISO-Autostep se retrouvent donc sans signaux de sortie sur les microphones de la *dark zone*.

La courbe de convergence de la partie « estimation des réponses impulsionnelles » est présentée en figure 9.5. Nous pouvons constater que l’algorithme a terminé de converger en 1000 s, avec deux phases de convergence. En effet, entre 400 s et 800 s, l’algorithme atteint un plateau et n’améliore plus les estimations. Cela correspond au moment où l’algorithme BACC-PM-FxLMS commence à mettre à jour les filtres de *sound zones*. En effet, sur la figure 9.6, les courbes de contraste et d’erreur de reproduction au cours du temps sont présentées. Nous pouvons ainsi voir qu’à partir de 400 s le contraste augmente et l’erreur de reproduction diminue.

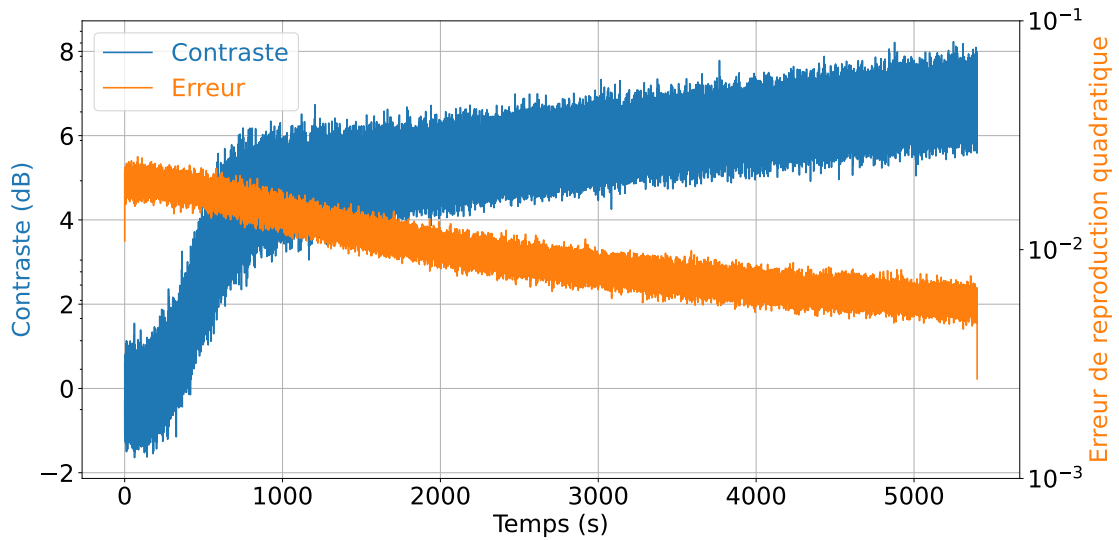


FIGURE 9.6 – Contraste et erreur de reproduction pour les algorithmes MISO-Autostep et BACC-PM-FxLMS.

Il est à noter des vitesses de convergence très lentes, et donc des temps de convergence très longs. Pour des raisons de temps de calcul, la simulation a été arrêtée à 1,6 h, et il faudrait attendre plusieurs heures pour obtenir des performances acceptables sur les *sound zones* (ce qui est bien trop long pour une utilisation pratique convenable). Cependant, cela montre qu’il est possible de générer des *sound zones* sans aucun *a priori* physique sur le système. La seule information nécessaire à l’initialisation du système est la taille des filtres, qui doivent englober l’entièreté des réponses impulsionnelles de la salle.

Le grand enjeu à venir est donc d’accélérer la convergence des algorithmes. Lors de l’installation du système, il est possible d’effectuer une mesure au préalable de toutes les

réponses impulsionnelles du système par une méthode classique, comme celle utilisant des signaux ESS présentée en section 2.1. Ces réponses impulsionnelles serviront alors de conditions initiales pour les algorithmes MISO-Autostep. De même, connaissant les réponses impulsionnelles dès le départ, il est possible d'utiliser l'algorithme BACC-PM pour trouver les premiers filtres de *sound zones* qui deviendront les conditions initiales pour l'algorithme BACC-PM-FxLMS. Cette technique permet de s'affranchir de la partie transitoire des différents filtres adaptatifs qui composent le système. Ce choix permet d'obtenir la figure 9.7, qui montre bien des performances identiques à l'algorithme BACC-PM dès le départ (seules les 60 premières secondes sont montrées).

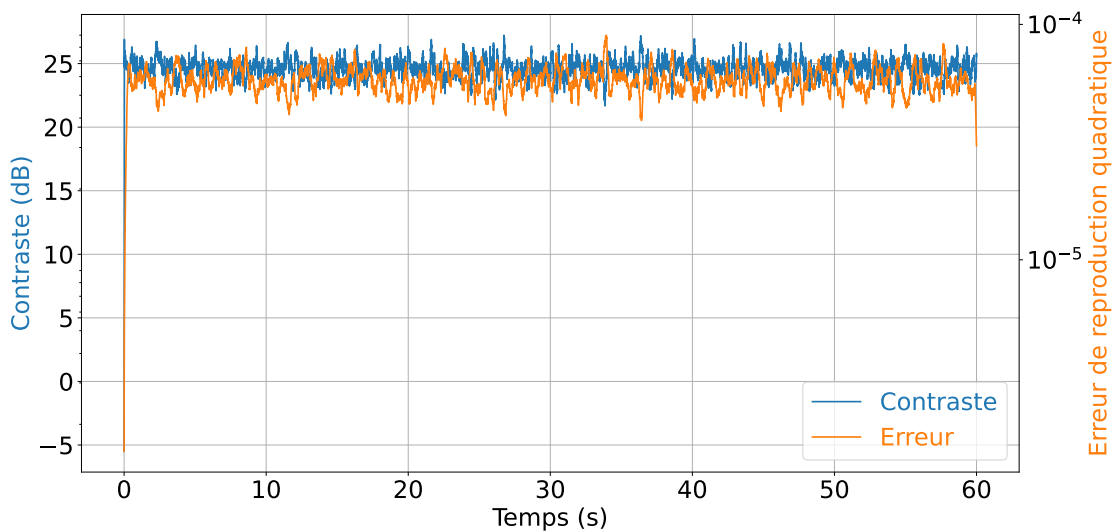


FIGURE 9.7 – Contraste et erreur de reproduction pour le Fx-LMS avec MISO-Autostep dans le cas de conditions initiales mesurées.

Une autre solution possible pourrait consister à changer de méthode d'optimisation pour l'algorithme de *sound zones*, par exemple en choisissant l'algorithme RLS, présenté en section 2.3.2, pour ses performances en termes de temps de convergence, au détriment du coût de calcul. Cependant, de nombreux algorithmes dits « rapides », permettent de diminuer les temps de calcul [102]-[106]. Une autre piste de travail pourrait être de diminuer l'ordre du système, en n'incluant pas la partie *dark zone* dans l'erreur (et donc en augmentant la taille de la matrice  $\mathbf{Q}(n)$ ), mais en la considérant sous la forme d'une contrainte sur les filtres, comme le propose l'algorithme RLS-DCD [107].

# CONCLUSION

---

L'objectif des *sound zones* est de reproduire différents contenus audio dans un même espace sans que les utilisateurs se gênent entre eux. Cette technologie a déjà été appliquée à l'aviation, l'automobile ou encore les *open-space*. Pour autant, toutes ces applications considèrent l'utilisateur immobile, et le système invariant dans le temps.

Or, des applications où les utilisateurs sont en mouvement existent : par exemple une pièce de vie commune, une visite de musée, etc. De plus, le système peut voir des variations au cours du temps. Par exemple, une modification de la température va modifier la vitesse du son et par conséquent les réponses impulsionnelles vont changer. Cependant, à l'exception des travaux de HU *et al.* [14], les algorithmes existants de *sound zones* ne permettent pas de déplacement quelconque une fois le système installé, et sont sensibles aux modifications du système.

Dans cette thèse, un algorithme adaptatif pour *sound zones* mobiles, décliné de BACC-PM, a été développé et testé dans différentes configurations, traitant à la fois de l'estimation des réponses impulsionnelles variant dans le temps et de la mise à jour des filtres de *sound zones* lorsque ces dernières changent dans le temps. Le manuscrit est séparé en deux parties : la première traite de l'estimation de réponses impulsionnelles de salle dans un contexte MISO et variant dans le temps, puis la seconde partie traite des algorithmes de *sound zones*. Le chapitre 9 quant à lui lie les deux parties et propose en perspective un algorithme de *sound zones* en boucle fermée qui estime les réponses impulsionnelles et génère les filtres de *sound zones* en même temps.

Dans une première partie, la problématique de l'estimation des réponses impulsionnelles variant dans le temps a été traitée. L'algorithme proposé MISO-Autostep permet alors d'estimer les réponses impulsionnelles du système au cours du temps, même dans le cas où le système varie dans le temps (à cause du déplacement de l'utilisateur, d'un changement de température, d'une modification dans la pièce). L'algorithme MISO-Autostep a l'avantage de pouvoir s'adapter à tout système de *sound zones*, qualité importante au regard de

---

la grande variété de réponses impulsionnelles rencontrées (courtes, longues, piquées, etc.), car il ne possède pas de paramètres à régler. De plus, comparé à l'algorithme existant MISO-LMS, le nouvel algorithme conserve un coût de calcul faible et la capacité de suivi de trajectoire tout en améliorant les performances de MISO-LMS. Sur les cas traités, autant en simulation qu'en mesure, l'erreur résiduelle après convergence est de 1 dB à 3 dB plus faible que MISO-LMS. L'algorithme MISO-Autostep étant très faiblement dépendant par rapport à ces méta-paramètres, il a toujours la même vitesse de convergence, au contraire de l'algorithme MISO-LMS dont la vitesse de convergence dépend de son paramètre de réglage  $\mu$  (ou  $\tilde{\mu}$  pour MISO-NLMS). Ainsi, s'il est possible de choisir une valeur de paramètre qui a une vitesse de convergence plus rapide que MISO-Autostep, l'erreur résiduelle sera *in fine* plus grande. Enfin, l'algorithme MISO-Autostep a été testé dans un cas d'estimation de réponses impulsionnelles de système MISO variant dans le temps, validant la possibilité d'étendre l'étude du cas LTI au cas LTV sans changer les valeurs des méta-paramètres de l'algorithme.

Dans la seconde partie, les filtres pour les *sound zones* ont été obtenus par descente de gradient et non plus par inversion directe. La méthode d'optimisation ainsi modifiée permet de mettre à jour les réponses impulsionnelles au cours du temps, y compris dans le cas où les zones sont mobiles. Deux algorithmes sont proposés : une descente de gradient classique BACC-PM-Gd et une descente de gradient conjugué BACC-PM-Gdc. Le premier a l'avantage de la simplicité algorithmique et du coût de calcul, quand le second a l'avantage d'une vitesse de convergence plus grande. Il a été montré que l'algorithme BACC-PM-Gd converge vers la solution trouvée par l'algorithme BACC-PM. Ainsi, des contrastes de 20 dB à 25 dB ont été obtenus dans une pièce, dans le cas statique ainsi qu'en mouvement pleine bande. En basses fréquences, entre 50 Hz et 500 Hz, des contrastes de plus de 30 dB ont été obtenus. L'erreur de reproduction quant à elle reste toujours, en statique comme en mouvement, inférieure à  $1 \times 10^{-4}$ . Les meilleurs résultats sont obtenus lorsque les zones sont immobiles, car dès qu'il y a mouvement de zone, les métriques diminuent, et tout particulièrement le contraste. Dans un cas pratique, cela correspond à un cas où l'utilisateur passe d'une position à une autre, toutes deux connues. Si dans les deux positions fixes, les algorithmes donnent des résultats proches des solutions statiques, lors du mouvement une dégradation du contraste est cependant à prévoir. Néanmoins, dans les deux états fixes, les performances peuvent être qualifiées de bonnes. Lors du mouvement, l'erreur de reproduction reste faible et donc l'utilisateur pourra continuer à

---

écouter son programme. Il sera par contre un peu plus gêné par le programme d'autres utilisateurs, puisque le contraste sera plus faible.

L'algorithme final, qui estime conjointement les réponses impulsionnelles et les filtres de *sound zones*, permet de réaliser des *sound zones* mobiles dans le cas de vitesse lente (ou quasi-statique). Dans le cas en boucle ouverte (i.e. les signaux d'entrée et l'erreur réelle ne sont pas pris en compte), les résultats ne sont pas satisfaisants, car les estimations des réponses impulsionnelles ne sont pas assez précises pour permettre une résolution correcte du problème inverse. Cependant, les vitesses de convergence (si les réponses impulsionnelles estimées sont proches de la réalité) sont grandes. Dans le cas où il y a peu de positions fixes prévues, il est possible de calibrer le système pour toutes ces positions et corriger les estimations avec les réponses impulsionnelles les plus proches pour obtenir des zones correctement contraintes.

L'algorithme a ainsi été modifié pour prendre en compte l'erreur réelle mesurée à l'aide des microphones des zones, le modifiant de fait en algorithme en boucle fermée. La connaissance partielle des réponses impulsionnelles est alors compensée par la connaissance de l'erreur réelle dans chaque zone. Il est toutefois à noter que les temps de convergence et de calcul restent encore longs et sa mise en place en temps réel nécessitera des travaux ultérieurs. De même, l'influence des paramètres et les valeurs des métriques dans un contexte mobile devront faire l'objet d'études perceptives.





# Annexes

# PROPRIÉTÉS DE CONVERGENCE DE L'ALGORITHME MISO-AUTOSTEP

---

Nous considérons le vecteur d'erreur  $\boldsymbol{\epsilon}_{n+1}^{(l)}$ , défini comme la différence entre la réponse impulsionnelle vraie  $\mathbf{h}_r^{(l)}$  et son estimation à l'instant  $n + 1$  (voir figure 3.1),

$$\boldsymbol{\epsilon}_{n+1}^{(l)} = \mathbf{h}_r^{(l)} - \mathbf{h}_{n+1}^{(l)}. \quad (\text{A.1})$$

Le signal de sortie bruité  $p_n$  s'écrit :

$$p_n = \sum_{l=1}^L \mathbf{h}_r^{(l)} \mathbf{x}_n^{(l)} + b_n, \quad (\text{A.2})$$

où  $b_n$  est un bruit blanc centré de variance  $\sigma_b^2$ . Nous obtenons alors :

$$\boldsymbol{\epsilon}_{n+1}^{(l)} = \mathbf{h}_r^{(l)} - \mathbf{h}_n^{(l)} - \boldsymbol{\mu}_n^{(l)} \circ e_n \mathbf{x}_n^{(l)}. \quad (\text{A.3})$$

Remplaçons l'expression de  $e_n$  (3.1) dans l'équation (A.3), soit :

$$\boldsymbol{\epsilon}_{n+1}^{(l)} = \mathbf{h}_r^{(l)} - \mathbf{h}_n^{(l)} - \boldsymbol{\mu}_n^{(l)} \circ \left( p_n - \sum_{j=1}^L \mathbf{h}_n^{(j)\top} \mathbf{x}_n^{(j)} \right) \mathbf{x}_n^{(l)}. \quad (\text{A.4})$$

Ensuite, remplaçons  $p_n$  par son expression de l'équation (A.2) pour obtenir :

$$\begin{aligned} \boldsymbol{\epsilon}_{n+1}^{(l)} &= \boldsymbol{\epsilon}_n^{(l)} - \boldsymbol{\mu}_n^{(l)} \circ \left[ \sum_{j=1}^L \left( \mathbf{h}_r^{(j)\top} - \mathbf{h}_n^{(j)\top} \right) \mathbf{x}_n^{(j)} \right] \mathbf{x}_n^{(l)} \\ &\quad - \boldsymbol{\mu}_n^{(l)} b_n \circ \mathbf{x}_n^{(l)}. \end{aligned} \quad (\text{A.5})$$

Rassemblons ensuite les termes liés à l'entrée  $l$  :

$$\begin{aligned}\boldsymbol{\epsilon}_{n+1}^{(l)} &= \boldsymbol{\epsilon}_n^{(l)} - \boldsymbol{\mu}_n^{(l)} \circ \left( \boldsymbol{\epsilon}_n^{(l)\top} \mathbf{x}_n^{(l)} \right) \mathbf{x}_n^{(l)} \\ &\quad - \boldsymbol{\mu}_n^{(l)} \circ \left( \sum_{\substack{j=1 \\ j \neq l}}^L \boldsymbol{\epsilon}_n^{(j)\top} \mathbf{x}_n^{(j)} \right) \mathbf{x}_n^{(l)} - \boldsymbol{\mu}_n^{(l)} b_n \circ \mathbf{x}_n^{(l)},\end{aligned}\tag{A.6}$$

et factorisons par  $\boldsymbol{\epsilon}_n^{(l)}$ . Après réagencement des termes, nous obtenons :

$$\begin{aligned}\boldsymbol{\epsilon}_{n+1}^{(l)} &= \left( \mathbf{I} - \boldsymbol{\mu}_n^{(l)} \circ \left( \mathbf{x}_n^{(l)} \mathbf{x}_n^{(l)\top} \right) \right) \boldsymbol{\epsilon}_n^{(l)} \\ &\quad - \sum_{\substack{j=1 \\ j \neq l}}^L \left( \boldsymbol{\mu}_n^{(l)} \circ \mathbf{x}_n^{(j)} \mathbf{x}_n^{(l)\top} \boldsymbol{\epsilon}_n^{(j)} \right) - \boldsymbol{\mu}_n^{(l)} \circ b_n \mathbf{x}_n^{(l)}.\end{aligned}\tag{A.7}$$

L'espérance de l'erreur s'écrit alors :

$$\begin{aligned}\mathbb{E}[\boldsymbol{\epsilon}_{n+1}^{(l)}] &= \left( \mathbf{I} - \mathbb{E}[\boldsymbol{\mu}_n^{(l)} \circ \left( \mathbf{x}_n^{(l)} \mathbf{x}_n^{(l)\top} \right)] \right) \mathbb{E}[\boldsymbol{\epsilon}_n^{(l)}] \\ &\quad - \sum_{\substack{j=1 \\ j \neq l}}^L \mathbb{E} \left( \boldsymbol{\mu}_n^{(l)} \circ \mathbf{x}_n^{(j)} \mathbf{x}_n^{(l)\top} \boldsymbol{\epsilon}_n^{(j)} \right) \\ &\quad - \mathbb{E}[\boldsymbol{\mu}_n^{(l)} \circ b_n \mathbf{x}_n^{(l)}].\end{aligned}\tag{A.8}$$

Considérons que les vecteurs  $\boldsymbol{\mu}$  sont asymptotiquement indépendants (et par conséquent décorrélés) des signaux d'entrée  $\mathbf{x}$  [108], soit  $\forall (i, j) \in [1, L]^2$ ,

$$\mathbb{E}[\boldsymbol{\mu}_n^{(i)} \mathbf{x}_n^{(j)}] = \mathbb{E}[\boldsymbol{\mu}_n^{(i)}] \mathbb{E}[\mathbf{x}_n^{(j)}].\tag{A.9}$$

Nous avons alors :

$$\begin{aligned}\mathbb{E}[\boldsymbol{\epsilon}_{n+1}^{(l)}] &= \left( \mathbf{I} - \mathbb{E}[\boldsymbol{\mu}_n^{(l)}] \circ \mathbb{E}[\mathbf{x}_n^{(l)} \mathbf{x}_n^{(l)\top}] \right) \mathbb{E}[\boldsymbol{\epsilon}_n^{(l)}] \\ &\quad - \sum_{\substack{j=1 \\ j \neq l}}^L \mathbb{E}[\boldsymbol{\mu}_n^{(l)}] \circ \mathbb{E}[\mathbf{x}_n^{(j)} \mathbf{x}_n^{(l)\top}] \mathbb{E}[\boldsymbol{\epsilon}_n^{(j)}] \\ &\quad - \mathbb{E}[\boldsymbol{\mu}_n^{(l)}] \circ \mathbb{E}[b_n \mathbf{x}_n^{(l)}].\end{aligned}\tag{A.10}$$

Le deuxième terme de droite de l'équation (A.10) est nul puisque les signaux d'entrées sont mutuellement indépendants les uns des autres. Le troisième terme est également nul

---

puisque le bruit additif est centré. Finalement, nous obtenons :

$$\mathbb{E}[\boldsymbol{\epsilon}_{n+1}^{(l)}] = (\mathbf{I} - \mathbb{E}[\boldsymbol{\mu}_n^{(l)}] \circ \mathbf{R}_x^{(l)}) \mathbb{E}[\boldsymbol{\epsilon}_n^{(l)}], \quad (\text{A.11})$$

avec  $\mathbf{R}_x^{(l)}$  la matrice d'autocorrélation de signal d'entrée  $l$  définie par  $\mathbf{R}_x^{(l)} = \mathbb{E}[\mathbf{x}_n^{(l)} \mathbf{x}_n^{(l)\top}]$ .

La matrice d'autocorrélation peut par ailleurs être factorisée suivant :

$$\mathbf{R}_x^{(l)} = \mathbf{Q}^{(l)} \boldsymbol{\Lambda}^{(l)} \mathbf{Q}^{(l)\top}, \quad (\text{A.12})$$

avec  $\mathbf{Q}^{(l)}$  la matrice orthonormale de vecteurs propres et  $\boldsymbol{\Lambda}^{(l)}$  la matrice diagonale contenant les valeurs propres. En notant  $\bar{\boldsymbol{\epsilon}}_n^{(l)} = \mathbf{Q}^{(l)\top} \boldsymbol{\epsilon}_n^{(l)}$ , nous déduisons que l'équation (A.11) équivaut à :

$$\mathbb{E}[\bar{\boldsymbol{\epsilon}}_{n+1}^{(l)}] = (\mathbf{I} - \mathbb{E}[\boldsymbol{\mu}_n^{(l)}] \circ \boldsymbol{\Lambda}^{(l)}) \mathbb{E}[\bar{\boldsymbol{\epsilon}}_n^{(l)}]. \quad (\text{A.13})$$

Ce résultat conduit à une condition de convergence sur  $\boldsymbol{\mu}_n^{(l)}$  :

$$0 < \mathbb{E}[\boldsymbol{\mu}_n^{(l)}] < \frac{2}{\lambda_{max}^{(l)}}, \quad (\text{A.14})$$

avec  $\lambda_{max}^{(l)}$  la plus grande valeur propre de la matrice d'autocorrélation  $\mathbf{R}_x^{(l)}$ . Sous cette condition, nous pouvons en déduire que :

$$\mathbb{E}[\boldsymbol{\epsilon}_n^{(l)}] \xrightarrow{n \rightarrow +\infty} 0 \implies \mathbb{E}[\mathbf{h}_n^{(l)}] \xrightarrow{n \rightarrow +\infty} \mathbf{h}_r^{(l)}. \quad (\text{A.15})$$

Ainsi, si la condition de convergence (A.14) est respectée, le filtre converge vers la solution optimale de Wiener.

# CHOIX DE LA GÉOMÉTRIE

---

En synthèse de champ, les systèmes audio nécessitent de nombreux haut-parleurs et microphones. Leur placement est donc un pré-requis avant l'utilisation du système. De nombreuses recherches ont été effectuées sur le placement de source pour optimiser leur position dans les systèmes de synthèse de champ [109]-[113].

L'article de KOYAMA *et al.* [114] présente notamment cinq algorithmes de positionnement des sources et/ou des microphones pour la synthèse de champ. Parmi les cinq algorithmes, un seul (le cinquième) propose de sélectionner les sources et microphones conjointement. Le premier algorithme traite du positionnement des sources, les trois suivants du placement des microphones. Il est à noter que les algorithmes développés ne prennent en compte qu'une seule *bright zone*.

Un travail préalable a été effectué dans le cadre d'un projet tutoré par des étudiants en Licence 3 Acoustique en 2022. En résultat préliminaire, les algorithmes 1 et 3 se sont avérés les plus performants en terme de contraste pour le choix de la géométrie. C'est donc sur ces deux algorithmes que nous nous sommes concentrés.

Il est à noter que dans cette partie l'étude a été effectuée dans le domaine fréquentiel. Nous noterons  $\mathbf{g}^{(l)} \in \mathbb{C}$  la fonction de transfert entre le haut-parleur  $l$  et tous les microphones, à une fréquence donnée. Sans autre indication, la fréquence de travail est  $f = 100$  Hz.

## B.1 Sélection des sources

Le premier algorithme permet de sélectionner les sources les mieux placées pour créer une *bright zone*. Dans notre cas, ayant deux zones à contrôler, nous définissons alors la zone à contraindre comme la somme des deux zones. Cet algorithme utilise l'orthogonalisation

---

de GRAM-SCHMIDT : il construit une base de vecteurs orthogonaux (issus des fonctions de transfert).

L'algorithme prend en entrée toutes les positions candidates, c'est-à-dire toutes les positions possibles pour les sources. Dans notre cas, les enceintes peuvent être disposées tout autour de la zone, sur le rail, définissant ainsi un carré de 3,2 m de côté. De plus, les sources sont espacées au minimum de 20 cm (largeur de l'enceinte). Les microphones sont fixes, sur deux antennes (une pour chaque zone), contenant neuf microphones chacune.

Le premier vecteur  $\mathbf{g}^l$  est choisi de telle manière à ce que la pression générée par ce haut-parleur est la plus proche possible de la pression cible choisie par le haut-parleur.

Ensuite, récursivement, l'algorithme va choisir un nouveau vecteur "fonction de transfert" de telle manière à ce que ce nouveau vecteur soit le plus orthogonal possible aux précédents vecteurs choisis. C'est-à-dire que la nouvelle fonction de transfert choisie soit la plus différente possible de toutes celles déjà sélectionnées. L'algorithme s'arrête lorsqu'il a sélectionné le bon nombre de haut-parleurs.

## B.2 Sélection des microphones

Cet algorithme considère que les sources sont fixes et il existe des positions candidates possibles pour les microphones.

Les haut-parleurs sont positionnés selon les positions sélectionnées par l'algorithme 1. Les positions candidates pour les microphones sont sur un quadrillage de 20 cm contenant 16 positions. L'algorithme doit choisir les neuf meilleures positions.

Cette fois-ci l'algorithme calcule une information mutuelle entre les microphones choisis et ceux restants. Récursivement, il sélectionne le microphone qui ajoute le moins d'information mutuelle (c'est-à-dire qui a une fonction de transfert la plus différente des autres possibles). L'algorithme s'arrête lorsqu'il a sélectionné le bon nombre de microphones.

Sur la figure B.1 nous remarquons que les haut-parleurs sélectionnés sont les plus proches des zones. Seuls deux haut-parleurs, parmi les vingt, sont éloignés dans l'angle inférieur droit.

ion des points de mesure dans le champ de haut

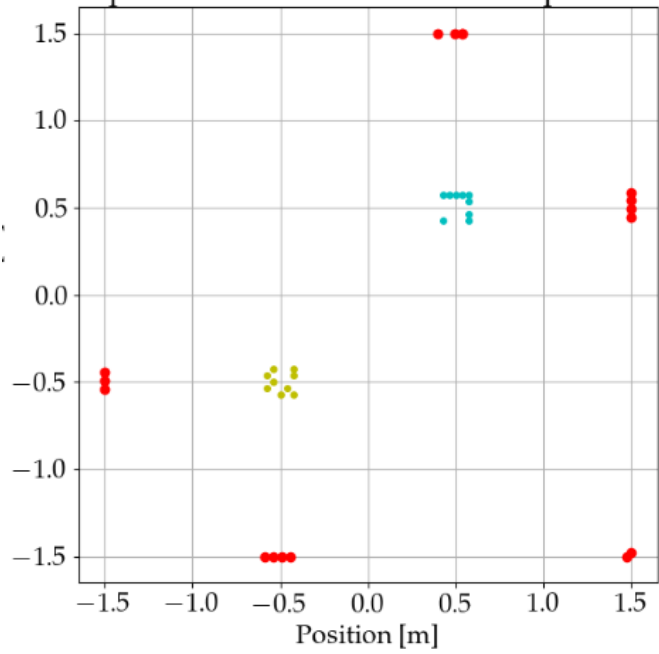


FIGURE B.1 – Géométrie optimisée





# BIBLIOGRAPHIE

---

- [1] A. J. BERKHOUT, D. de VRIES et P. VOGEL, « Acoustic control by wave field synthesis », *The Journal of the Acoustical Society of America*, t. 93, 5, p. 2764-2778, mai 1993.
- [2] O. KIRKEBY et P. A. NELSON, « Reproduction of plane wave sound fields », *The Journal of the Acoustical Society of America*, t. 94, 5, p. 2992-3000, nov. 1993.
- [3] W. J. TROTT, « Underwater-Sound-Transducer calibration from nearfield data », *The Journal of the Acoustical Society of America*, t. 36, 8, p. 1557-1568, août 1964.
- [4] Y. SUDO, « Active noise cancelling apparatus », *The Journal of the Acoustical Society of America*, t. 101, 5, p. 2426, 1997.
- [5] O. KIRKEBY, P. A. NELSON, F. ORDUNA-BUSTAMANTE et H. HAMADA, « Local sound field reproduction using digital signal processing », *The Journal of the Acoustical Society of America*, t. 100, 3, p. 1584-1593, sept. 1996.
- [6] W. F. DRUYVESTYEN et J. GARAS, « Personal Sound », *Journal of the Audio Engineering Society*, t. 45, 9, p. 685-701, 1997.
- [7] J.-W. CHOI et Y.-H. KIM, « Generation of an acoustically bright zone with an illuminated region using multiple sources », *The Journal of the Acoustical Society of America*, t. 111, 4, p. 1695-1700, avr. 2002.
- [8] T. BETLEHEM, W. ZHANG, M. A. POLETTI et T. D. ABHAYAPALA, « Personal sound zones : delivering interface-free audio to multiple listeners », *IEEE Signal Processing Magazine*, t. 32, 2, p. 81-91, mars 2015.
- [9] J. CHEER et S. ELLIOTT, « Design and implementation of a personal audio system in a car cabin », *Proceedings of Meetings on Acoustics*, t. 19, 1, p. 055 009, mai 2013.
- [10] S. J. ELLIOTT et M. JONES, « An active headrest for personal audio », *The Journal of the Acoustical Society of America*, t. 119, 5, p. 2702-2709, mai 2006.

- 
- [11] S. J. ELLIOTT, J. CHEER, H. MURFET et K. R. HOLLAND, « Minimally radiating sources for personal audio », *The Journal of the Acoustical Society of America*, t. 128, 4, p. 1721-1728, oct. 2010.
- [12] F. HEUCHEL, D. CAVIEDES NOZAL et F. T. ZGERKVIST, « Sound field control for reduction of noise from outdoor concerts », *Journal of the Audio Engineering Society*, 10107, oct. 2018.
- [13] Y. HUANG, J. BENESTY et J. CHEN, « Identification of acoustic MIMO systems : Challenges and opportunities », *Signal Processing*, t. 86, 6, p. 1278-1295, juin 2006.
- [14] M. HU, L. SHI, H. ZOU, M. G. CHRISTENSEN et J. LU, « Sound zone control with fixed acoustic contrast and simultaneous tracking of acoustic transfer functions », *The Journal of the Acoustical Society of America*, t. 153, 5, p. 2538, mai 2023.
- [15] M. B. MOLLER et J. OSTERGAARD, « A moving horizon framework for sound zones », *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, t. 28, p. 256-265, 2020.
- [16] L. VINDROLA, M. MELON, J.-C. CHAMARD et B. GAZENGEL, « Use of the filtered-x least-mean-squares algorithm to adapt personal sound zones in a car cabin », *The Journal of the Acoustical Society of America*, t. 150, 3, p. 1779-1793, sept. 2021.
- [17] T. YOSHIOKA et T. NAKATANI, « Generalization of multi-channel linear prediction methods for blind MIMO impulse response shortening », *IEEE Transactions on Audio, Speech, and Language Processing*, t. 20, 10, p. 2707-2720, 2012.
- [18] I. KODRASI et S. DOCLO, « Joint dereverberation and noise reduction based on acoustic multi-channel equalization », *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, t. 24, 4, p. 680-693, 2016.
- [19] M. HU, J. XUE et J. LU, « Online multi-channel secondary path modeling in active noise control without auxiliary noise », *The Journal of the Acoustical Society of America*, t. 146, 4, p. 2590-2595, oct. 2019.
- [20] S. MAKINO, « Stereophonic acoustic echo cancellation : An overview and recent solutions. », *Acoustical Science and Technology*, t. 22, 5, p. 325-333, 2001.
- [21] J. BORISH et J. B. ANGELL, « An efficient algorithm for measuring the impulse response using pseudorandom noise », *Journal of the Audio Engineering Society*, t. 31, 7, p. 478-488, août 1983.

- 
- [22] A. FARINA, « Simultaneous measurement of impulse response and distortion with a swept-sine technique », *AES 108th convention*, p. 1-24, fév. 2000.
- [23] P. GUILLAUME, R. PINTELOU et J. SCHOUKENS, « Parametric identification of two-port models in the frequency domain », *IEEE Transactions on Instrumentation and Measurement*, t. 41, p. 233-238, 1992.
- [24] Y. TABATA, T. EBIHARA, H. OGASAWARA, N. WAKATSUKI, K. ZEMPO et K. MIZUTANI, « Underwater acoustic communication using orthogonal signal division multiplexing with windowing », *Japanese Journal of Applied Physics*, t. 61, SG, SG1033, mai 2022.
- [25] S. S. HAYKIN, *Adaptive filter theory*, 5th ed. Upper Saddle River, NJ, USA : Pearson, 2014.
- [26] R. SUTTON, « Adapting bias by gradient descent : An incremental version of Delta-Bar-Delta », in *Proceedings of the 10th National Conference on Artificial Intelligence*, 1992, p. 171-176.
- [27] A. R. MAHMOOD, R. S. SUTTON, T. DEGRIS et P. M. PILARSKI, « Tuning-free step-size adaptation », in *2012 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, IEEE, mars 2012, p. 2121-2124.
- [28] P. GUIDORZI, L. BARBARESI, D. D'ORAZIO et M. GARAI, « Impulse responses measured with MLS or Swept-Sine signals applied to architectural acoustics : an in-depth analysis of the two methods and some case studies of measurements inside theaters », *Energy Procedia*, t. 78, p. 1611-1616, nov. 2015.
- [29] S. GUY-BART, E. JEAN-JACQUES et A. DOMINIQUE, « Comparison of different impulse response measurement techniques », *Journal of the Audio Engineering Society*, t. 50, p. 249-262, 4 avr. 2002.
- [30] A. NOVAK, P. LOTTON et L. SIMON, « Synchronized swept-sine : Theory, application, and implementation », *Journal of the Audio Engineering Society*, t. 63, 10, p. 786-798, 2015.
- [31] P. THUNE et G. ENZNER, « Trends in adaptive MISO system identification for multichannel audio reproduction and speech communication », in *2013 8th International Symposium on Image and Signal Processing and Analysis (ISPA)*, Trieste, Italy : IEEE, sept. 2013, p. 767-772.

- 
- [32] P. MAJDAK, « Multiple exponential sweep method for fast measurement of Head-Related Transfer Functions », *Journal of the Audio Engineering Society*, t. 55, 7, 2007.
- [33] S. KANAI, K. MATSUI et S. ADACHI, « Identification input design for simultaneous estimation of head-related transfer functions », in *2014 Proceedings of the SICE Annual Conference (SICE)*, Sapporo, Japan : IEEE, sept. 2014, p. 1673-1678.
- [34] R. GREEN et T. NGUYEN, « Signal Processing and its Applications », in *Signal Processing for Active Control*, Elsevier, 2001.
- [35] S. SONG, J.-S. LIM, S. BAEK et K.-M. SUNG, « Gauss Newton variable forgetting factor recursive least squares for time varying parameter tracking », *Electronics Letters*, t. 36, 11, p. 988, 2000.
- [36] A. BORDES, L. BOTTOU et P. GALLINARI, « SGD-QN : careful quasi-Newton stochastic gradient descent », *Journal of Machine Learning*, t. 10, p. 1737-1754, 2009.
- [37] B. WIDROW, *Adaptive signal processing*. Philadelphia, PA : Pearson Education, 2016.
- [38] D. BISMOR, K. CZYZ et Z. OGOROWSKI, « Review and comparison of variable step-size LMS algorithms », *The International Journal of Acoustics and Vibration*, t. 21, 1, 2016.
- [39] T. SHAN et T. KAILATH, « Adaptive algorithms with an automatic gain control feature », *IEEE Transactions on Circuits and Systems*, t. 35, 1, p. 122-127, 1988.
- [40] G. PAGÈS, R. LONGO, L. SIMON et M. MELON, « Online adaptive identification of multichannel systems for audio applications », *The Journal of the Acoustical Society of America*, t. 155, 1, p. 229-240, jan. 2024.
- [41] S. BAGHERI, M. THILL, P. KOCH et W. KONEN, « Online adaptable learning rates for the game Connect-4 », *IEEE Transactions on Computational Intelligence and AI in Games*, t. 8, 1, p. 33-42, 2016.
- [42] J. CASEBEER, N. J. BRYAN et P. SMARAGDIS, « Auto-DSP : learning to optimize acoustic echo cancellers », in *2021 IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA)*, 2021, p. 291-295.
- [43] M. BRUNEAU, *Fundamentals of acoustics*. Wiley-ISTE, mars 2013.

- 
- [44] J. SNOEK, H. LAROCHELLE et R. P. ADAMS, « Practical bayesian optimization of machine learning algorithms », in *Proceedings of the 25th International Conference on Neural Information Processing Systems - Volume 2*, 2012, p. 2951-2959.
- [45] E. A. LEHMANN, A. M. JOHANSSON et S. NORDHOLM, « Reverberation-time prediction method for room impulse responses simulated with the image-source model », in *2007 IEEE Workshop on Applications of Signal Processing to Audio and Acoustics*, oct. 2007, p. 159-162.
- [46] J.-H. CHANG, C.-H. LEE, J.-Y. PARK et Y.-H. KIM, « A realization of sound focused personal audio system using acoustic contrast control », *The Journal of the Acoustical Society of America*, t. 125, 4, p. 2091-2097, avr. 2009.
- [47] M. SIMÓN GÁLVEZ, S. ELLIOTT et J. CHEER, « Personal audio loudspeaker array as a complementary TV sound system for the hard of hearing », *IEICE Transactions on Fundamentals of Electronics Communications and Computer Sciences*, t. 97, p. 1824-1831, sept. 2014.
- [48] W.-S. GAN, J. YANG et T. KAMAKURA, « A review of parametric acoustic array in air », *Applied Acoustics*, t. 73, 12, p. 1211-1219, 2012.
- [49] J. KUUTTI, J. LEIWO et R. SEPPONEN, « Local control of audio environment : A review of methods and applications », *Technologies*, t. 2, 1, p. 31-53, fév. 2014.
- [50] F. ZANGENEH-NEJAD et R. FLEURY, « Active times for acoustic metamaterials », *Reviews in Physics*, t. 4, p. 100 031, nov. 2019.
- [51] J. AHRENS, R. RABENSTEIN et S. SASCHA, « The theory of wave field synthesis revisited », *Journal of the Audio Engineering Society*, 7358, mai 2008.
- [52] S. KOYAMA, K. FURUYA, Y. HIWASAKI et Y. HANEDA, « Analytical approach to wave field reconstruction filtering in spatio-temporal frequency domain », *IEEE Transactions on Audio, Speech, and Language Processing*, t. 21, 4, p. 685-696, avr. 2013.
- [53] T. OKAMOTO, « Mode-matching-based sound field recording and synthesis with circular double-layer arrays », *Applied Sciences*, t. 8, 7, p. 1084, juill. 2018.
- [54] M. A. POLETTI, « Three-Dimensional surround sound systems based on spherical harmonics », *Journal of the Audio Engineering Society*, t. 53, 11, p. 1004-1025, 2005.

- 
- [55] J. AHRENS et S. SPORS, « An analytical approach to sound field reproduction using circular and spherical loudspeaker distributions », *Acta Acustica united with Acustica*, t. 94, 6, p. 988-999, nov. 2008.
- [56] M. SHIN, S. Q. LEE, F. M. FAZI et al., « Maximization of acoustic energy difference between two spaces », *The Journal of the Acoustical Society of America*, t. 128, 1, p. 121-131, juill. 2010.
- [57] N. YANAGIDATE, J. CHEER, S. ELLIOTT et T. TOI, « Car cabin personal audio : Acoustic contrast with limited sound differences », *Proceedings of the AES International Conference*, t. 2014, août 2014.
- [58] P. COLEMAN, P. JACKSON, M. OLIK et J. A. PEDERSEN, « Optimizing the planarity of sound zones », *Journal of the Audio Engineering Society*, 5-1, sept. 2013.
- [59] M. SIMÓN GÁLVEZ, S. ELLIOTT et J. CHEER, « Time domain optimization of filters used in a loudspeaker array for personal audio », *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, t. 23, 11, p. 1869-1878, nov. 2015.
- [60] M. POLETTI, « An investigation of 2-D multizone surround sound systems », in *Audio Engineering Society Convention 125*, oct. 2008.
- [61] J. BENESTY, M. CHRISTENSEN et J. R. JENSEN, *Signal enhancement with Variable Span Linear Filters*. jan. 2016, t. 7.
- [62] J. R. JENSEN, J. BENESTY et M. G. CHRISTENSEN, « Noise reduction with Optimal Variable Span Linear Filters », *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, t. 24, 4, p. 631-644, avr. 2016.
- [63] T. LEE, J. K. NIELSEN, J. R. JENSEN et M. G. CHRISTENSEN, « A unified approach to generating Sound Zones using Variable Span Linear Filters », in *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, Calgary, AB : IEEE, avr. 2018, p. 491-495.
- [64] T. ABE, S. KOYAMA, N. UENO et H. SARUWATARI, « Amplitude matching for multizone sound field control », *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, t. 31, p. 656-669, 2023.
- [65] J. CHENG, H. GAO, K. PAN, X. FENG et Y. SHEN, « Multi-objective optimization for generation of personal sound zone », *The Journal of the Acoustical Society of America*, t. 153, 1, p. 149-158, jan. 2023.

- 
- [66] Y. J. WU et T. D. ABHAYAPALA, « Spatial multizone soundfield reproduction : Theory and design », *IEEE Transactions on Audio, Speech, and Language Processing*, t. 19, 6, p. 1711-1720, août 2011.
- [67] W. ZHANG, T. D. ABHAYAPALA, T. BETLEHEM et F. M. FAZI, « Analysis and control of multi-zone sound field reproduction using modal-domain approach », *The Journal of the Acoustical Society of America*, t. 140, 3, p. 2134-2144, sept. 2016.
- [68] W. ZHANG, P. SAMARASINGHE, H. CHEN et T. ABHAYAPALA, « Surround by sound : A review of spatial audio recording and reproduction », *Applied Sciences*, t. 7, 5, p. 532, mai 2017.
- [69] W. JIN, W. B. KLEIJN et D. VIRETTE, « Multizone soundfield reproduction using orthogonal basis expansion », in *2013 IEEE International Conference on Acoustics, Speech and Signal Processing*, Vancouver, BC, Canada : IEEE, mai 2013, p. 311-315.
- [70] K. BAYKANER, P. COLEMAN, R. MASON et al., « The relationship between target quality and interference in sound zone », *Journal of the Audio Engineering Society*, t. 63, p. 78-89, jan. 2015.
- [71] J. RÄMÖ, L. CHRISTENSEN, S. BECH et S. JENSEN, « Validating a perceptual distraction model using a personal two-zone sound system », Boston, Massachusetts, 2017, p. 050 003.
- [72] J. FRANCOMBE, R. MASON, M. DEWHIRST et S. BECH, « Elicitation of attributes for the evaluation of audio-on-audio interference », *The Journal of the Acoustical Society of America*, t. 136, 5, p. 2630-2641, nov. 2014.
- [73] K. R. BAYKANER, « Predicting the perceptual acceptability of auditory interference for the optimisation of sound zones », thèse de doct., University of Surrey (United Kingdom)., Guildford, 2014, p. 307.
- [74] J. DONLEY, C. RITZ et W. B. KLEIJN, « Multizone soundfield reproduction with privacy- and quality-based speech masking filters », *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, t. 26, 6, p. 1041-1055, juin 2018.



- 
- [75] N. de KOELJER, M. B. MØLLER, J. MARTINEZ, P. MARTÍNEZ-NUEVO et R. C. HENDRIKS, « Block-based perceptually adaptive sound zones with reproduction error constraints », *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, t. 32, p. 3090-3100, 2024.
- [76] L. VINDROLA, M. MELON, J.-C. CHAMARD, B. GAZENGEL et G. PLANTIER, « Personal Sound Zones : a comparison between frequency and time domain formulations in a transportation context », *147th AES Convention (New York)*, oct. 2019.
- [77] O. KIRKEBY et P. A. NELSON, « Digital Filter Design for Inversion Problems in Sound Reproduction », *Journal of the Audio Engineering Society*, t. 47, 7/8, p. 583-595, 1999.
- [78] S. J. ELLIOTT, J. CHEER, J.-W. CHOI et Y. KIM, « Robustness and regularization of personal audio systems », *IEEE Transactions on Audio, Speech, and Language Processing*, t. 20, 7, p. 2123-2133, sept. 2012.
- [79] Y. CAI, M. WU, L. LIU et J. YANG, « Time-domain acoustic contrast control design with response differential constraint in personal audio systems », *The Journal of the Acoustical Society of America*, t. 135, 6, juin 2014.
- [80] D. H. M. SCHELLEKENS, M. B. MOLLER et M. OLSEN, « Time domain acoustic contrast control implementation of sound zones for low-frequency input signals », in *2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, Shanghai : IEEE, mars 2016, p. 365-369.
- [81] J. CADAVID, M. B. MØLLER, C. S. PEDERSEN, S. BECH, T. VAN WATERSCHOOT et J. ØSTERGAARD, « Performance of low-frequency sound zones with very fast room impulse response measurements », *The Journal of the Acoustical Society of America*, t. 155, 1, p. 757-768, jan. 2024.
- [82] V. MOLÉS-CASES, S. J. ELLIOTT, J. CHEER, G. PIÑERO et A. GONZALEZ, « Weighted pressure matching with windowed targets for personal sound zones », *The Journal of the Acoustical Society of America*, t. 151, 1, p. 334-345, jan. 2022.
- [83] L. SHI, T. LEE, L. ZHANG, J. K. NIELSEN et M. GROSBOLL CHRISTENSEN, « A fast reduced-rank Sound Zone control algorithm using the conjugate gradient method », in *ICASSP 2020 - 2020 IEEE International Conference on Acoustics,*

- 
- Speech and Signal Processing (ICASSP)*, Barcelona, Spain : IEEE, mai 2020, p. 436-440.
- [84] L. SHI, T. LEE, L. ZHANG, J. K. NIELSEN et M. G. CHRISTENSEN, « Generation of personal Sound Zones with physical meaningful constraints and conjugate gradient method », *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, t. 29, p. 823-837, 2021.
- [85] J. BRUNNSTRÖM, T. VAN WATERSCHOOT et M. MOONEN, « Signal-to-interference-plus-noise ratio based optimization for sound zone control », *IEEE Open Journal of Signal Processing*, t. 4, p. 257-266, 2023.
- [86] M. B. MOLLER, J. K. NIELSEN, E. FERNANDEZ-GRANDE et S. K. OLESEN, « On the influence of transfer function noise on Sound Zone control in a room », *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, t. 27, 9, p. 1405-1418, sept. 2019.
- [87] J. ZHANG, L. SHI, M. G. CHRISTENSEN, W. ZHANG, L. ZHANG et J. CHEN, « CGMM-Based Sound Zone generation using robust pressure matching with ATF perturbation constraints », *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, t. 31, p. 3331-3345, 2023.
- [88] G. ROUSSEL, « Contributions à la mise au point de méthodes adaptatives de reproduction de champs sonores multi-zone pour les auditeurs en mouvement : Sound zones pour auditeurs en mouvement », thèse de doct., Université du Maine, juill. 2019.
- [89] M. OLSEN et M. B. MØLLER, « Sound Zones : On the effect of ambient temperature variations in feed-forward systems », in *Audio Engineering Society Convention 142*, mai 2017.
- [90] M. B. MØLLER, J. MARTINEZ et J. ØSTERGAARD, « Reduced complexity for sound zones with subband block adaptive filters and a loudspeaker line array », *The Journal of the Acoustical Society of America*, t. 155, 4, p. 2314-2326, avr. 2024.
- [91] D. MOREAU, B. CAZZOLATO, A. ZANDER et C. PETERSEN, « A review of virtual sensing algorithms for active noise control », *Algorithms*, t. 1, 2, p. 69-99, nov. 2008.

- 
- [92] T. PHAM VU et H. LISSEK, « Low frequency sound field reconstruction in a non-rectangular room using a small number of microphones », *Acta Acustica*, t. 4, 2, p. 5, 2020.
- [93] H. W. KUHN et A. W. TUCKER, *Nonlinear programming*. Berkeley, Calif. : University of California Press, 1951, p. 481-492.
- [94] J.-H. CHANG et F. JACOBSEN, « Sound field control with a circular double-layer array of loudspeakers », en, *The Journal of the Acoustical Society of America*, t. 131, 6, p. 4518-4525, juin 2012.
- [95] F. R. GANTMACHER, *Théorie des matrices*. Jacques Gabay, 1990.
- [96] D. G. LUENBERGER et Y. YE, *Linear and nonlinear programming* (International series in operations research & management science). Springer International Publishing, juin 2015.
- [97] H. AKAIKE, « On a successive transformation of probability distribution and its application to the analysis of the optimum gradient method », *Annals of the Institute of Statistical Mathematics*, t. 11, 1, p. 1-16, oct. 1959.
- [98] J. R. SHEWCHUK, « An introduction to the conjugate gradient method without the agonizing pain », USA, rapp. tech., 1994.
- [99] R. C. BROWNING, E. A. BAKER, J. A. HERRON et R. KRAM, « Effects of obesity and sex on the energetic cost and preferred speed of walking », *Journal of Applied Physiology*, t. 100, 2, p. 390-398, 2006.
- [100] O. TOBIAS et R. SEARA, « Leaky-FXLMS algorithm : stochastic analysis for Gaussian data and secondary path modeling error », *IEEE Transactions on Speech and Audio Processing*, t. 13, 6, p. 1217-1230, nov. 2005.
- [101] P. NELSON, H. HAMADA et S. ELLIOTT, « Adaptive inverse filters for stereophonic sound reproduction », *IEEE Transactions on Signal Processing*, t. 40, 7, p. 1621-1632, 1992.
- [102] J. CIOFFI et T. KAILATH, « Fast, recursive-least-squares transversal filters for adaptive filtering », *IEEE Transactions on Acoustics, Speech, and Signal Processing*, t. 32, 2, p. 304-337, 1984.
- [103] V. DJIGAN, « Low Complexity RLS Adaptive Filters », in *2022 24th International Conference on Digital Signal Processing and its Applications (DSPA)*, 2022, p. 1-5.

- 
- [104] A. CARINI et E. MUMOLO, « A numerically stable fast RLS algorithm for adaptive filtering and prediction based on the UD factorization », *IEEE Transactions on Signal Processing*, t. 47, 8, p. 2309-2313, 1999.
- [105] N. XIE, H. WANG et X. LI, « Fast Adaptive Equalizer Algorithm with Low Complexity », *Chinese Journal of Electronics*, t. 20, 3, p. 550-552, 2011.
- [106] S. QIAO, « Fast adaptive RLS algorithms : a generalized inverse approach and analysis », *IEEE Transactions on Signal Processing*, t. 39, 6, p. 1455-1459, 1991.
- [107] V. H. NASCIMENTO et Y. V. ZAKHAROV, « RLS Adaptive Filter With Inequality Constraints », *IEEE Signal Processing Letters*, t. 23, 5, p. 752-756, 2016.
- [108] M. O. B. SAEED, « A unified analysis approach for LMS-based variable step-size algorithms », *Arabian Journal for Science and Engineering*, t. 42, 7, p. 2809-2816, juill. 2017.
- [109] S. KOYAMA, T. AMAKASU, N. UENO et H. SARUWATARI, « Amplitude Matching : majorization–minimization algorithm for sound field control only with amplitude constraint », in *ICASSP 2021 - 2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, Toronto, ON, Canada : IEEE, juin 2021, p. 411-415.
- [110] T. NISHIDA, N. UENO, S. KOYAMA et H. SARUWATARI, « Region-restricted sensor placement based on gaussian process for sound field estimation », *IEEE Transactions on Signal Processing*, t. 70, p. 1718-1733, 2022.
- [111] K. ARIGA, T. NISHIDA, S. KOYAMA, N. UENO et H. SARUWATARI, « Mutual-information-based sensor placement for spatial sound field recording », in *ICASSP 2020 - 2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, Barcelona, Spain : IEEE, mai 2020, p. 166-170.
- [112] J. XU, K. CHEN et Y. LI, « Wave-domain optimization of secondary source placement free from information of error sensor positions », in *ICASSP 2021 - 2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, Toronto, ON, Canada : IEEE, juin 2021, p. 166-170.
- [113] K. ARIKAWA, S. KOYAMA et H. SARUWATARI, « Spatial active noise control method based on sound field interpolation from reference microphone signals », in *ICASSP 2023 - 2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, Rhodes Island, Greece : IEEE, juin 2023, p. 1-5.

- 
- [114] S. KOYAMA, G. CHARDON et L. DAUDET, « Optimizing source and sensor placement for sound field control : An overview », *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, t. 28, p. 696-714, 2020.



**Titre :** Zones d'écoute personnalisées mobiles par approches adaptatives

**Mot clés :** bulle sonore, synthèse de champ, audio, optimisation, filtrage adaptatif

**Résumé :** La thèse aborde la création de *sound zones* (zones d'écoute personnalisées) mobiles par des approches adaptatives. Les méthodes en usage pour la création de *sound zones* visent à résoudre conjointement la reproduction sonore dans une zone et la minimisation de l'énergie du signal reproduit dans l'autre zone, à partir d'un réseau de haut-parleurs. La thèse se divise en deux parties : l'estimation des réponses impulsionnelles et les *sound zones* mobiles. L'objectif de cette thèse est de créer deux zones de l'espace dont le champ acoustique est contrôlé, ces zones pouvant bouger dans l'espace au cours du temps. Dans la première partie, l'estimation des réponses impulsionnelles du système est détaillée, prérequis nécessaire pour les algorithmes de *sound zones*. A par-

tir des méthodes adaptatives existantes pour l'estimation de systèmes multi-entrées multi-sorties variant dans le temps, une nouvelle méthode appliquée à l'acoustique et MISO est présentée. Cette méthode, appelée MISO-Autostep, permet d'estimer les réponses impulsionnelles au cours du temps sans avoir de paramètres à ajuster finement. Dans la seconde partie, l'algorithme de *sound zones* BACC-PM est réécrit sous forme récursive. Cette capacité à mettre à jour les coefficients des filtres au cours du temps ouvre la voie à s'adapter à des changements temporels dans la géométrie du système. Enfin, des résultats préliminaires sont présentés avec l'utilisation conjointe des deux algorithmes adaptatifs dans le cas d'une modification brusque de la géométrie du système.

**Title:** Personalised mobile sound zones using adaptive approaches

**Keywords:** sound zone, wave field synthesis, audio, optimisation, adaptive filtering

**Abstract:** The thesis deals with the creation of mobile *sound zones* using adaptive approaches. The methods in use for the creation of *sound zones* aim to jointly resolve the sound reproduction in one zone and the minimisation of the energy of the signal reproduced in the other zone, from an array of loudspeakers. The thesis is divided into two parts: the estimation of impulse responses and moving sound zones. The aim of this thesis is to create two zones in space with a controlled sound field, which can move in space over time. In the first part, the estimation of the system's impulse responses is detailed, a necessary prerequisite for sound zone algorithms.

Based on existing adaptive methods for estimating time-varying multi-input, multi-output systems, a new method applied to acoustics and MISO is presented. This method, called MISO-Autostep, makes it possible to estimate impulse responses over time without having to fine-tune any parameters. In the second part, the BACC-PM sound zone algorithm is rewritten in recursive form. This ability to update the filter coefficients over time opens up the possibility of adapting to temporal changes in the system geometry. Finally, preliminary results are presented with the joint use of the two adaptive algorithms in the case of an abrupt change in the system geometry.