

RÉPUBLIQUE ALGÉRIENNE DÉMOCRATIQUE ET POPULAIRE
Ministère de l'Enseignement Supérieur et de la Recherche Scientifique

École Nationale Polytechnique



Département d'Electronique
Laboratoire de Communication et de Conversion Photovoltaïque

*Mémoire de projet de fin d'études
pour l'obtention du diplôme d'ingénieur d'état en Electronique*

THÈME :

**Conception et réalisation d'un réseau de microphones
pour l'acquisition sonore :
exploitation des techniques de filtrage spatial.**

BENDERMEL Qasem

DJEMAI Mohamed

Sous la direction de

Pr. Adel BELOUHRANI

PhD. Mourad ADNANE

Présenté et soutenu publiquement le 17 Juin devant le jury composé de :

Président	Pr. M. MEHENNI	(ENP)
Examineur	Pr. R. AKSAS	(ENP)
Promoteur	Pr. A. BELOUHRANI	(ENP)
Co-Promoteur	Dr. M. ADNANE	(ENP)

ENP 2017

RÉPUBLIQUE ALGÉRIENNE DÉMOCRATIQUE ET POPULAIRE
Ministère de l'Enseignement Supérieur et de la Recherche Scientifique

École Nationale Polytechnique



Département d'Electronique
Laboratoire de Communication et de Conversion Photovoltaïque

*Mémoire de projet de fin d'études
pour l'obtention du diplôme d'ingénieur d'état en Electronique*

THÈME :

**Conception et réalisation d'un réseau de microphones
pour l'acquisition sonore :
exploitation des techniques de filtrage spatial.**

BENDERMEEL Qasem

DJEMAI Mohamed

Sous la direction de

Pr. Adel BELOUHRANI

PhD. Mourad ADNANE

Présenté et soutenu publiquement le 17 Juin devant le jury composé de :

Président	Pr. M. MEHENNI	(ENP)
Examineur	Pr. R. AKSAS	(ENP)
Promoteur	Pr. A. BELOUHRANI	(ENP)
Co-Promoteur	Dr. M. ADNANE	(ENP)

ENP 2017

Dédicace

Je dédie ce travail à mes très chères mère et grande mère, dont le sacrifice, la tendresse, l'amour, la patience, le soutien, l'aide et l'encouragement sont l'essentiel de ma réussite. Sans elles je ne serai pas à ce stade aujourd'hui.

A mes cousins Mustapha et Foudil pour leur soutien continue durant mon parcours.

A Brahim, Mounir, Oussama, et Kamel.

A Rahim, Mehdi et Madjid pour leur présence.

M. DJEMAI

A mes chers parents

Je dédie ce modeste travail à ceux sans qui mon existence n'aurait pas eu lieu, ceux sans qui le bonheur n'aurait été qu'une illusion, ceux qui ont fait de ma vie un comble de joie. Aucune dédicace ne saurait être assez éloquente pour exprimer ce que vous méritez pour tous les sacrifices que vous avez consentis pour mon bien être, à mes parents les êtres les plus chers dans ma vie. Que dieu vous préserve et vous accorde la santé et une longue vie et votre bénédiction m'accompagnera toujours.

A mes chers frères et soeurs

Les mots ne suffisent guère pour exprimer l'attachement, l'amour et l'affection que je porte pour vous. Je vous souhaite un avenir plein de joie, de bonheur, de réussite et de sérénité

A mes amis

Un énorme merci à Billal, Kadiro, Ilyes, Hichem et Wedi avec qui j'ai pu gardé le sourire au pires moments de ma vie.

A mon binome

Djemai, avec qui j'ai eu le plaisir de travailler dans une bonne ambiance.

Q. BENDERMEEL

Remerciements

Nous remercions Dieu le tout puissant de nous avoir donné la force et le courage pour réaliser ce travail.

Les travaux présentés dans ce mémoire ont été effectués au sein de l'École Nationale Polytechnique.

Ce travail que nous présentons a été effectué sous la direction de Mr *M. ADNANE*, PhD et chef de département de l'électronique à l'École Nationale Polytechnique, qui ont suivi ce travail, pour leur orientation pédagogique dans l'élaboration de ce mémoire.

Nous tenons aussi à remercier Mr *C. LARBES*, Docteur à l'École Nationale Polytechnique et Mr *Samir*, dirigeant de Orbit, pour nous avoir aidé avec le matériel nécessaire.

Nous tenons à remercier Mr *M. MEHENNI*, Professeur à l'École Nationale Polytechnique, pour l'honneur qu'il nous fait de présider le jury de notre soutenance.

Que Mr *R. AKSAS*, Professeur à l'École Nationale Polytechnique, soit convaincu de notre sincère reconnaissance pour avoir accepté d'examiner et de critiquer ce mémoire.

Enfin, nous tenons à remercier tous les personnes qui ont contribué à ma réussite tout au long de mon parcours d'étude.

ملخص

معالجة شبكة الهوائيات هو مجال دراسة مخصص لمعالجة الإشارات الواردة من مجموعة من الهوائيات واستخراج المعلومات المرغوبة. ولقد لعبت دورا هاما في التطبيقات الشائعة مثل الرادار والسونار والاتصالات اللاسلكية. وقد تم طرح العديد من خوارزميات معالجة شبكة الهوائيات التي تتأقلم مع الظروف المحيطة بها على مدى العقود الماضية. هذه الخوارزميات، على العموم، تقدم حلا وسطا بين الأداء والتعقيد الحسابي المطلوب. في هذه الأطروحة، نحن نركز على تطوير خوارزميات معالجة مجموعة ميكروفونات في تطبيق تكوين الحزم. نحن نقدم مجموعة متنوعة من التقنيات، ولكل منها مزاياها وعيوبها، مع تقديم مقارنة بين هاته الخوارزميات باستخدام كل من البيانات المركبة والبيانات الحقيقية.

الكلمات المفتاحية: معالجة مجموعة ميكروفونات، تكوين الحزم، مجموعة خطية متناسقة.

Abstract

Array processing is an area of study devoted to processing signals received from an antenna array and extracting information of interest. It has played an important role in widespread applications like radar, sonar, and wireless communications. Numerous adaptive array processing algorithms have been reported in the literature in the last several decades. These algorithms, in a general view, exhibit a trade-off between performance and required computational complexity. In this thesis, we focus on the development of microphone array processing algorithms in the application of beamforming. We present a variety of methods, each with its advantages and its inconvenients, providing a comparison with both synthesized and real world data.

Key words: microphone array signal processing, beamforming, uniform linear array.

Résumé

Le traitement d'antennes est un domaine d'étude consacré au traitement des signaux reçus d'un réseau d'antennes et à l'extraction d'informations utiles. Il a joué un rôle important dans des applications répandues comme le radar, le sonar et les communications sans fil. De nombreux algorithmes de traitement d'antennes adaptative ont été publiés au cours des dernières décennies. Ces algorithmes, en général, présentent un compromis entre la performance et la complexité informatique requise. Dans ce mémoire, nous nous concentrons sur le développement d'algorithmes de traitement de réseau de microphones pour l'application de la formation de voies. Nous présentons une variété de méthodes, chacune avec ses avantages et ses inconvénients, fournissant une comparaison avec les données synthétisées ainsi qu'avec les données réelles.

Mots clés : traitement de réseau de microphones, formation de voies, réseau linéaire uniforme.

Table des matières

Table de figures

Abréviations et symboles

Introduction générale	10
1 Théorie	12
1.1 Le son	12
1.1.1 La pression sonore et le décibel	12
1.1.2 Les ondes sonores	13
1.1.3 Réverbération	14
1.1.4 L'hypothèse de propagation des ondes planes	14
1.2 Microphones	14
1.2.1 Directivité et réponse fréquentielle	15
1.2.2 Le réseau de microphones	16
1.2.3 Réponse spatiale d'un réseau de microphones	18
1.2.4 L'espacement entre microphones	19
2 Beamforming	21
2.1 Le filtre à retard numérique	23
2.1.1 Exemple	25
2.2 retard et somme (Delay-and-sum)	26
2.3 Minimum de Variance Réponse Non Distordue (MVDR)	29
2.3.1 MVDR fréquentiel	31
2.4 Formateurs de voies à filtre à réponse impulsionnelle finie (FIR)	34
2.4.1 L'idée de base de cet algorithme	34
2.4.2 Le modèle FIR	35
2.4.3 Le formateur de voies de Frost	37

2.4.4	Le formateur de voies Linearly Constrained Minimum Variance (LCMV)	38
2.4.5	Anulleur des lobes secondaires généralisé (Generalized sidelobe canceller)	40
2.4.6	L'algorithme LCMV avec la structure GSC	41
2.4.7	Le filtre LCMV avec le modèle réverbérant	43
2.4.8	Le filtre LCMV avec le modèle spatio-temporel	45
2.4.9	La méthode LS (Least Squares)	47
2.5	Les formateurs de voies adaptatifs	49
2.5.1	Estimation de la matrice de covariance	49
2.5.2	L'implémentation direct	49
2.5.3	L'implémentation LMS	50
2.5.4	L'implémentation GSC	51
2.5.5	L'implémentation de l'algorithme TF-GSC	52
2.6	Les performances des algorithmes	57
2.6.1	Le gain du réseau	58
2.6.2	performance de MVDR	59
2.6.3	performance de l'algorithme retard et somme	59
2.6.4	performance de LCMV	60
2.7	Conclusion	60
3	Tests et résultats	61
3.1	Les résultats de simulation	61
3.1.1	La performance de l'algorithme Delay-and-sum	62
3.1.2	La performance de l'algorithme MVDR dans le domaine fréquentiel	64
3.1.3	La performance de l'algorithme de frost	64
3.2	Les résultats expérimentaux	66
3.2.1	Le montage des composants	66
3.2.2	Le test du système d'acquisition	69
3.2.3	Les résultats expérimentaux des algorithmes	71
3.3	Conclusion	74
	Conclusion	75
	Bibliographie	76

Table des figures

1.1	Caractéristiques de l'onde.	13
1.2	Diagramme de directivité.	15
1.3	Réseau de 1020 microphones conçu par un équipe de MIT.	16
1.4	Réseau commercialisé de 24 microphones pour la téléconférence.	16
1.5	Réseau de 24 microphones pour la téléconférence.	16
1.6	réseau de microphones linéaires.	17
1.7	Réseau linéaire.	18
1.8	Réponse spatiale, $N=4$, $d=0.2m$, $f=1kHz$	19
1.9	Diagramme de directivité d'un réseau de microphones linéaire avec quatre éléments, en fonction de la direction horizontale θ , avec un échantillonnage spatial critique, $d = \lambda/2$ (gauche), et avec des effets d'alias pour λ (à droite).	20
2.1	réponse spatiale d'un réseau à bande étroite opérant dans une large bande.	23
2.2	La structure de filtre à retard numérique.	24
2.3	Le signaux retardés d'une portion de signal échantillonné avec une fréquence de 8kHz.	25
2.4	Le signaux alignés.	26
2.5	Retard-et-somme.	27
2.6	Réponse d'un réseau de 10 microphones, $\theta = 90$, $d = 5$ cm, $f = 3.5$ kHz.	28
2.7	Le bloc de traitement de MVDR fréquentiel.	31
2.8	Bloc de traitement de filtre a réponse impulsionnelle finie.	35
2.9	L'implémentation de l'algorithme GSC.	41
2.10	Le diagramme de directivité du filtre LS.	48
2.11	La structure de l'algorithme TF-GSC.	55
3.1	Illustration de l'environnement de simulation, le réseau de microphones reçoit 3 sources de 3 directions différentes : $\theta = 0$ degrés, $\theta = 30$ degrés, $\theta = -30$ degrés.	62
3.2	le signal d'entrée, le signal de sortie et le signal désiré du formateur de voies Delay-and-sum avec $N = 4$ microphones.	63

3.3	le signal d'entrée, le signal de sortie et le signal désiré du formateur de voies Delay-and-sum avec $N = 10$ microphones.	63
3.4	le signal d'entrée, le signal de sortie et le signal désiré du formateur de voies MVDR fréquentiel avec $N = 4$ microphones.	64
3.5	le signal d'entrée, le signal de sortie et le signal désiré du formateur de voies Frost avec $N = 4$ microphones.	65
3.6	le signal d'entrée, le signal de sortie et le signal désiré du formateur de voies Frost avec $N = 10$ microphones.	65
3.7	l'environnement des expériences, le réseau de microphones reçoit : une source de la direction $\theta = 0$ degrés, une source de la direction $\theta = 30$ degrés.	66
3.8	Schéma bloc du montage du système d'acquisition.	67
3.9	Le réseau de microphones réalisé, avec $N = 4$ microphones et $d = 5$ cm.	67
3.10	La carte NI USB-6002.	68
3.11	La source est dans la direction $\theta = 0$ degrés, la distance entre la source et le centre du réseau est 1m.	69
3.12	Les signaux du réseau de microphones réslisé avec 4 microphones, la source est une source monochromatique ($f=1000\text{Hz}$), la fréquence d'échantillonnage est 12500Hz , $\theta = 90^\circ$	70
3.13	La source est dans la direction $\theta = 90^\circ$, la distance entre la source et le centre du réseau est 1m.	70
3.14	Les signaux du réseau de microphones réslisé avec 4 microphones, la source est une source monochromatique ($f=1000\text{Hz}$), la fréquence d'échantillonnage est 12500Hz , $\theta = 90$ degrés.	71
3.15	Le signal d'entrée et le signal de sortie avec l'algorithme Delay and Sum $N = 4$	72
3.16	Le signal d'entrée et le signal de sortie avec l'algorithme Frost direct $N = 4$	73
3.17	Le signal d'entrée et le signal de sortie avec l'algorithme Frost LMS $N = 4$	73
3.18	Le signal d'entrée et le signal de sortie avec l'algorithme Frost LMS et avec la structure GSC $N = 4$	74

Abréviations et symboles

$s(t)$: Source sonore (parole).

$n(t)$: Le bruit.

$x_i(t)$: signal reçu à la i ème microphone.

$y(t)$: La sortie de formateur de voies.

ν : La vitesse de son dans le vide (343 m/s).

λ : La longueur d'onde.

τ_i : Le retard entre le i ème microphone et la référence.

θ_i : L'angle d'arrivé du i ème signal.

d : La distance entre deux microphone adjacents.

DOA : Direction d'arrivé.

MVDR : Minimum Variance Distorsionless Response.

FIR : Finite Impulse Response.

GSC : Generalized Sidelobe Cancellor.

LCMV : Linearly Constrained Minimim Variance.

LS : Least Square.

LMS : Least Mean Square.

Introduction générale

Dans notre monde toujours aussi mondialisé, et avec la maturité des technologies de traitement de la parole, une nouvelle génération d'applications d'acquisition de la parole apparaît. La société moderne qui continue à éprouver une grande envie d'améliorer l'interactivité entre des individus, en fournissant à l'utilisateur le meilleur confort, la flexibilité, la qualité et la facilité d'emploi. Les microphones classiques doivent être proches de l'utilisateur en tout temps, ce qui oblige l'utilisateur à porter le microphone ou à le faire déplacer avec lui. Actuellement, il existe des moyens d'utiliser de nombreux microphones pour créer des modèles de faisceau qui se concentrent sur l'orateur dans une pièce. L'utilisation des microphones orientables électriquement concerne principalement les téléconférences et les applications.

Dans le développement et la conception d'un système audio de conférence plusieurs aspects de la qualité du son doivent être considérés. D'une part, le bruit et les perturbations provenant des haut-parleurs ainsi que de l'environnement doivent être éliminés du signal des microphones pour éviter les bruits et les échos. D'autre part, les signaux utiles doivent être améliorés. Les matériaux doivent être choisis et la construction doit être telle que les vibrations soient minimisées, et la mise en place des microphones et des éléments de haut-parleurs devrait faciliter le traitement du signal. De plus, étant donné que les participants d'une conférence peuvent être positionnés loin des microphones, à des endroits qui peuvent ou pas être connus, on doit utiliser des algorithmes pour améliorer le ou les signaux vocaux désirés sont nécessaires. Cette dernière question et les méthodes d'amélioration du signal employant plusieurs microphones sont l'objet de ce travail.

On peut utiliser plus d'un microphone en combinant les sorties des microphones pour améliorer le ou les signaux désirés. Un tel ensemble de microphones forme un réseau de microphones dont les signaux reçus sont traités et additionnés pour obtenir un meilleur signal. En général, le traitement des signaux en réseau se concentre sur trois tâches : la détection, l'estimation de la direction d'arrivée (DOA) et le filtrage spatial (Beamforming) [1].

Les systèmes de filtrage spatial sont couramment utilisés pour améliorer la qualité d'un signal reçu. Des applications typiques peuvent être trouvées dans les communications radio, le traitement du signal radar, l'acoustique sous-marine et l'acquisition de la parole pour les systèmes de téléconférence et mains libres. Le filtrage spatial permet de focaliser le réseau dans une certaine direction d'où le signal souhaité se propage, et atténue simultanément des signaux d'interférences ou le bruit se propageant depuis toutes les autres directions. Ceci est plus détaillé dans le chapitre Beamforming.

Le formateur de faisceau le plus simple, connu sous le nom de formateur de faisceau à retard et somme (delay-and-sum), est construit comme son nom l'indique, en retardant chaque entrée de capteur en fonction de la direction du signal d'arrivée et en ajoutant ensuite les entrées ensemble [2]. Ce faisant, le signal d'intérêt est ajouté en phase, tandis que le bruit et les autres interférences sont ajoutés en opposition de phase, ce qui améliore le signal recherché. Il s'agit d'un formateur de faisceau fixe où tous les capteurs sont également pondérés. Cette technique simple fonctionne assez bien pour les signaux d'une fréquence spécifique ou des signaux à bande étroite où la réponse en fréquence est raisonnablement constante, mais pour les signaux à large bande tels que la parole, une pondération dépendante de la fréquence est nécessaire [1].

Indépendamment de quelle technique de formation de faisceau est utilisée, on part du principe que la direction d'arrivée du signal utile est donnée. Par conséquent, pour utiliser la formation de faisceaux, la direction d'arrivée doit être connue a priori ou doit être estimée.

Ce travail a pour but d'explorer des solutions pour améliorer le signal de la parole reçu en utilisant plusieurs microphones, c'est-à-dire une solution à réseau de microphones, arrivant à une configuration microphone mains libres. Il est intéressant d'étudier comment de telles méthodes fonctionnent dans diverses conditions et comment des paramètres différents tels que la géométrie de réseau et le type de microphones utilisés peuvent affecter la performance.

Chapitre 1

Théorie

Ce chapitre introduit le lecteur à quelques notions de base du son et des caractéristiques pertinentes des microphones. La partie principale du chapitre résume les connaissances nécessaires sur le traitement statistique du signal numérique, y compris le traitement d'antennes.

1.1 Le son

L'acoustique comprend tous les types d'ondes mécaniques se propageant à travers les gaz, les liquides et les solides. Le son peut être défini comme de telles ondes ou comme la sensation causée par la stimulation des organes auditifs avec les vibrations de ces ondes.[3]

1.1.1 La pression sonore et le décibel

La pression sonore est la pression acoustique d'une onde sonore par rapport à un certain niveau de référence p_0 , généralement de $20 \mu\text{Pa}$ dans l'air. Comme la pression acoustique varie de très faibles à très grandes valeurs, le niveau de pression acoustique (SPL) est mesuré à l'aide d'une unité logarithmique, à savoir le décibel (dB). Pour toute puissance physique, le niveau de puissance W par rapport à W_0 peut être exprimé par :

$$L_W(dB) = 10 \log_{10} \frac{W}{W_0}. \quad (1.1)$$

La puissance acoustique est proportionnelle au carré de pression acoustique. Le niveau de pression acoustique p , par rapport à p_0 , peut être exprimé par :

$$L_p(dB) = 10 \log_{10} \frac{p^2}{p_0^2} = 20 \log_{10} \frac{p}{p_0}. \quad (1.2)$$

L'utilisation des rapports et de l'échelle logarithmique simplifie non seulement les calculs de SPL, mais aussi les rend plus intuitifs. A tout niveau de pression, une augmentation de la pression d'un facteur de deux correspond à une augmentation de 6 dB.

Le niveau de référence de $20 \mu\text{Pa}$ correspond au seuil de bruit perceptible pour l'oreille humaine [3]. Le rapport signal à bruit (SNR) est une mesure couramment exprimée en dB car elle est un rapport entre la puissance du signal et la puissance du bruit.

1.1.2 Les ondes sonores

Un signal sonore peut être représenté en notation complexes. Sous l'hypothèse d'une périodicité, une onde complexe peut à son tour être représentée par des composantes sinusoïdales plus simples. Par conséquent, toute onde sonore périodique peut être modélisée comme une somme d'ondes sinusoïdales [3].

Pour une onde périodique, soit T la période de cette onde. La distance parcourue par l'onde pendant ce temps est la longueur d'onde, λ , liée à la période par $\lambda = \nu T$, où ν est la vitesse du son dans le vide.

La fréquence, f , est le nombre de périodes par seconde (Hertz) et est donc liée à la période par la relation $f = \frac{1}{T}$. L'amplitude de crête, A , est la valeur absolue maximale du signal. On notera que d'autres dérivations d'amplitude apparaissent dans la littérature, par ex. Amplitude crête à crête qui mesure la différence entre les valeurs de signal les plus élevées et les valeurs les plus faibles. Les caractéristiques utilisées ici sont présentées graphiquement dans la figure 1.1.

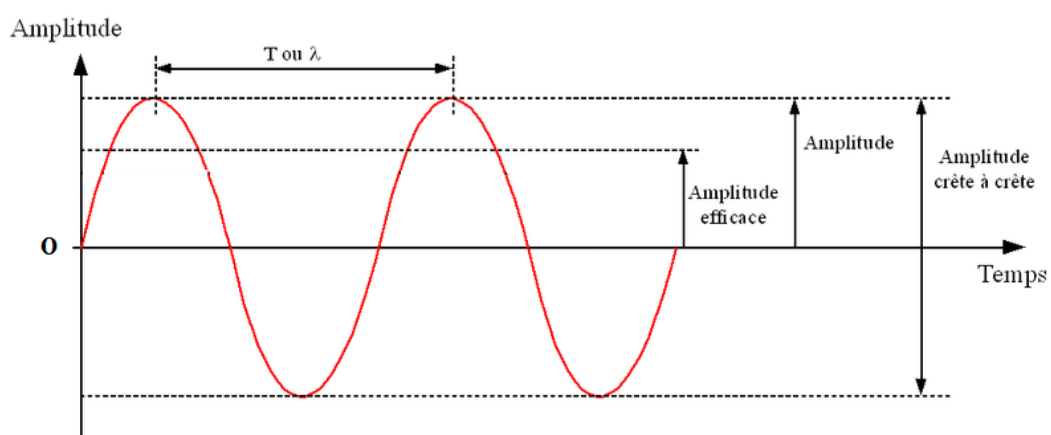


FIGURE 1.1 – Caractéristiques de l'onde.

1.1.3 Réverbération

Quand un son est produit, il sera encore audible peu de temps après. Un mot sera entendu dans toute la pièce après avoir été prononcé et le son d'une note de piano va s'attarder même après que la touche ait été pressée. Cet effet est appelé réverbération et est dû aux nombreuses réflexions des ondes sonores qui se propagent dans la salle et atteignent le récepteur à différents moments. Le temps qu'il faut avant que le son ne soit inaudible dépend des caractéristiques de la pièce. Avec des matériaux absorbants, le son décroît rapidement alors que pour les surfaces réfléchissantes le temps de réverbération s'étend.

Le temps de réverbération est défini comme le temps, en secondes, qu'il faut pour que le son dans une pièce s'affaiblisse de 60 dB. Du fait que les matériaux absorbants présentent des caractéristiques différentes pour des fréquences différentes, le temps de réverbération dépend de la fréquence [3].

1.1.4 L'hypothèse de propagation des ondes planes

Comme il est courant dans la littérature sur le traitement de signal provenant d'un réseau, la source utile sera supposée être dans le champ lointain. Puisque les ondes sonores se propagent d'une manière omnidirectionnelle dans l'air, à courtes distances, les ondes qui arrivent sur le réseau seront sous forme d'une sphère. Cependant, à des distances importantes, les ondes atteignant le réseau apparaîtront planes, c'est-à-dire que le lieu des points émis simultanément sera un plan et non pas une sphère. La source est généralement considérée comme étant dans le champ proche lorsque la distance de l'interlocuteur au réseau est de la même dimension que le réseau ou la taille d'ouverture (distance entre deux microphones). Pour être considérée dans le champ lointain, la distance entre le réseau et la source doit donc être plus grande que l'ouverture.

L'hypothèse de champ lointain facilite la modélisation et les calculs dans le traitement de signal. Cependant, lorsque la distance du réseau au interlocuteur est inconnue et qu'il est rarement possible d'estimer, l'hypothèse du champ lointain est la plus raisonnable à prendre en pratique [4].

1.2 Microphones

Dans le traitement du signal d'un réseau de microphones, la géométrie et la taille du réseau ainsi que le nombre de microphones ont un impact sur les performances du réseau. En outre, le type de microphones utilisés est un facteur important. Dans ce travail, deux types différents

de microphones peuvent être utilisés, omnidirectionnels et unidirectionnels, pour lesquels les différentes caractéristiques seront discutées dans ce paragraphe.

1.2.1 Directivité et réponse fréquentielle

La directivité ou la réponse directionnelle d'un microphone décrit comment le son issu de différentes directions est capté par le microphone. Il est communément présenté dans un diagramme polaire du gain par rapport à l'angle, et les différents types de microphones sont appelés suivant la forme de leur directivité. Les deux types mentionnés dans ce travail sont présentés dans la figure 1.2. Un microphone omnidirectionnel répond également au son provenant de toutes les directions. La directivité d'un microphone dépend de la fréquence, mais un microphone omnidirectionnel optimal devrait avoir un gain égal sur toutes les directions. La réponse unidirectionnelle montrée à la figure 1.2 est un microphone cardioïde typique, nommé d'après son motif de gain en forme de cœur. Les microphones unidirectionnels répondent au son provenant de l'avant du microphone, tandis que la lecture des côtés et de l'arrière est réduite. La réponse des microphones unidirectionnels varie avec la fréquence et est généralement étroite aux fréquences élevées alors que ces microphones sont pratiquement omnidirectionnels aux basses fréquences. En outre, la réponse en fréquence des microphones unidirectionnels varie également avec la distance entre le microphone et la source sonore ; plus la distance est longue, plus la lecture des basses fréquences est une caractéristique connue sous le nom d'effet de proximité [5].

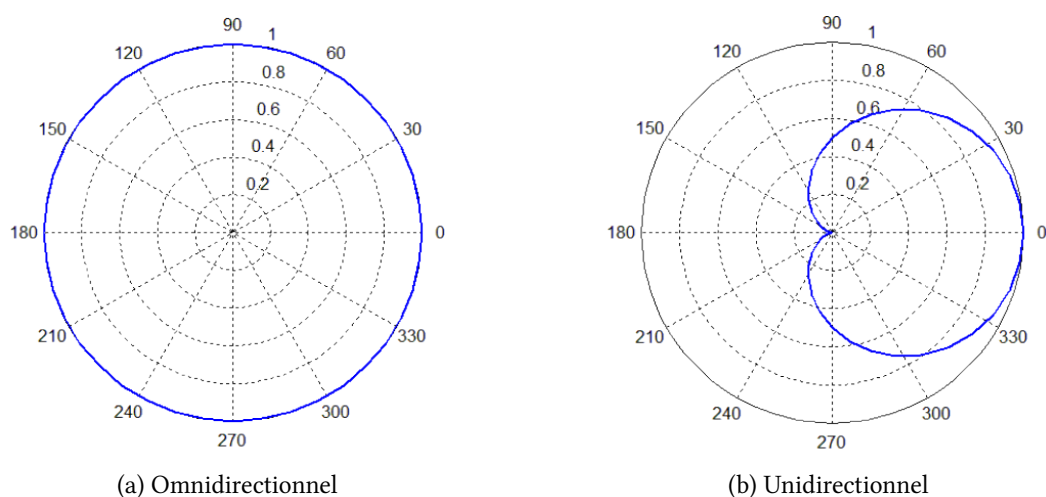


FIGURE 1.2 – Diagramme de directivité.

1.2.2 Le réseau de microphones

L'étude et la mise en œuvre des réseaux de microphones sont survenues il y a plus de 30 ans. Grâce à la recherche et aux développements expérimentaux, le domaine a mûri au point que la technologie basée sur le réseau a maintenant une applicabilité immédiate à un certain nombre de systèmes actuels et un vaste potentiel pour l'amélioration des produits existants et la création de futurs appareils, voir les figures 1.4, 1.5, 1.3.

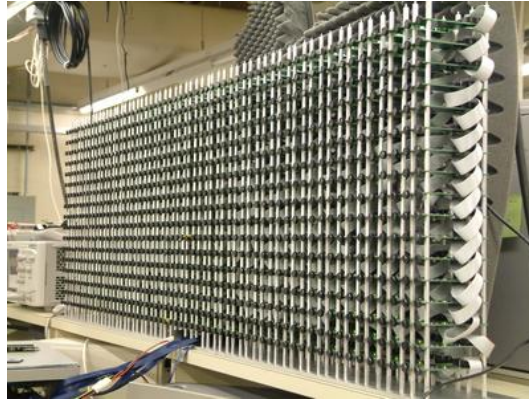


FIGURE 1.3 – Réseau de 1020 microphones conçu par un équipe de MIT.

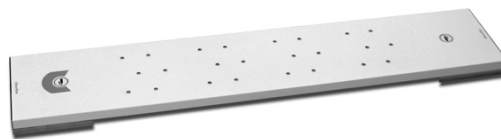


FIGURE 1.4 – Réseau commercialisé de 24 microphones pour la téléconférence.



FIGURE 1.5 – Réseau de 24 microphones pour la téléconférence.

Le réseau de capteurs acoustiques se compose d'un ensemble de capteurs acoustiques placés à différents endroits afin de recevoir les signaux. La figure 1.6 montre la disposition d'un réseau de microphone linéaire composé de M microphones avec K signaux incidents provenant des angles d'arrivée θ_K .

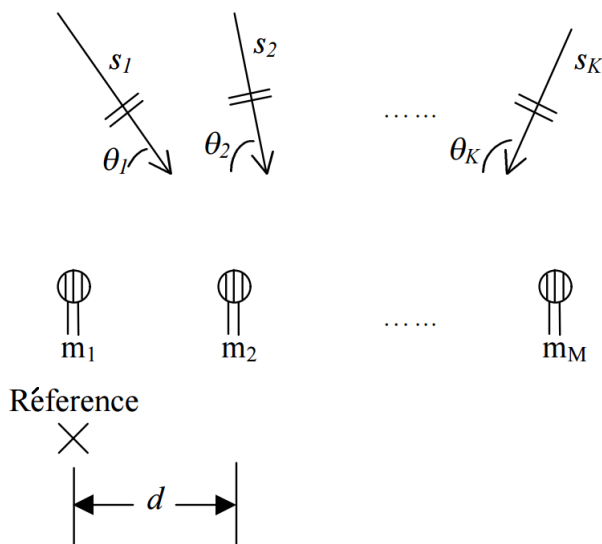


FIGURE 1.6 – réseau de microphones linéaires.

Dans cette analyse, on suppose que les ondes incidentes sont des ondes planes. Chaque microphone recevra les signaux, s_k , mais ceux-ci seront des versions retardées ou avancées dans le temps. Il faut donc un point de référence pour le système de coordonnées afin que nous puissions calculer la phase de chaque signal, nous choisirons le microphone 1 comme origine. La distance entre deux microphones adjacents est d (voir la figure 1.6).

La distance parcourue par le son planaire de chaque source s_k à chacun des microphones par rapport à la distance au microphone m_1 sera $d_i \cos \theta_k$. Le retard correspondant τ_i à chaque microphone est alors :

$$\tau_i = \frac{d_i \cos \theta_k}{\nu} \quad (1.3)$$

Où ν est la vitesse du son dans le vide ($343m/s$) et d_i est la distance entre la référence (l'origine) et l' i ème microphone.

L'entrée à l' i ème microphone du signal s_k est donc :

$$x_i(t) = s_k(t - \tau_i) \quad (1.4)$$

dans un scénario à bande étroite, pour des petites variations autour de λ , le retard peut être exprimé par un simple déphasage :

$$x_i(t) = s_k(t)e^{-j\omega_0\tau_i} \quad (1.5)$$

qui est équivalent à

$$x_i(t) = s_k(t)e^{-jkd \cos \theta_k} \quad (1.6)$$

où $k = \frac{2\pi}{\lambda}$ est le nombre d'onde.

La représentation de (1.5) est plus simple à utiliser que la représentation (1.4). Maintenant si on veut voir une représentation générale on doit introduire tous les signaux, et inclure aussi le bruit à chaque microphone :

$$x_i(t) = \sum_{k=1}^K s_k(t)e^{-jkd \cos \theta_k} + n_i(t) \quad (1.7)$$

1.2.3 Réponse spatiale d'un réseau de microphones

Un calcul simple peut être utilisé pour déterminer la sensibilité d'un réseau de microphones pour les signaux provenant d'une direction particulière. La figure 1.7 montre un réseau de quatre microphones. Deux microphones adjacents sont séparés par une distance d (en mètres).

L'équation 1.8 calcule le gain du réseau lorsqu'un seul signal $x(t)$ est présent, avec une seule fréquence f et un angle d'arrivée θ . c désigne la vitesse du son et N le nombre de microphones.

$$\frac{y(t)}{x(t)} = \frac{1}{N} \sum_{i=0}^{N-1} \exp\left(\frac{2\pi f}{\nu} i d \cos \theta\right). \quad (1.8)$$

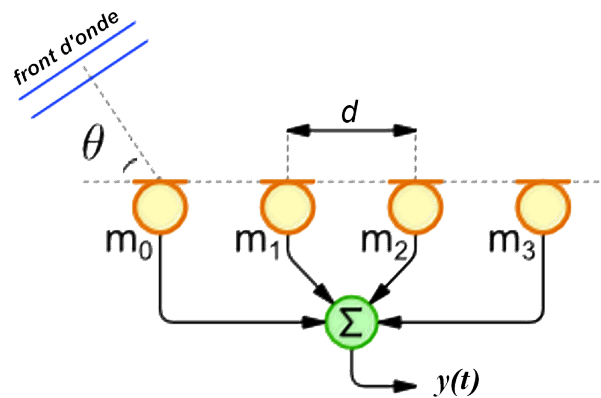


FIGURE 1.7 – Réseau linéaire.

L'équation fait quelques hypothèses ; la source de signal est suffisamment éloignée du réseau pour lequel le front d'onde est effectivement plat, il n'y a pas non plus de prise en compte de l'atténuation du signal lorsqu'elle se déplace de la source vers les microphones.

Le gain du réseau est tracé sur la figure 1.8. La sortie est normalisée à la sortie qui serait reçue à partir de microphone de référence, c'est à droite. Par conséquent, à un angle de 90 degrés (la cible est en face du réseau), l'amplitude de sortie équivaut à un microphone omnidirectionnel, ce qui donne un gain de 1 (ou 0 dB).

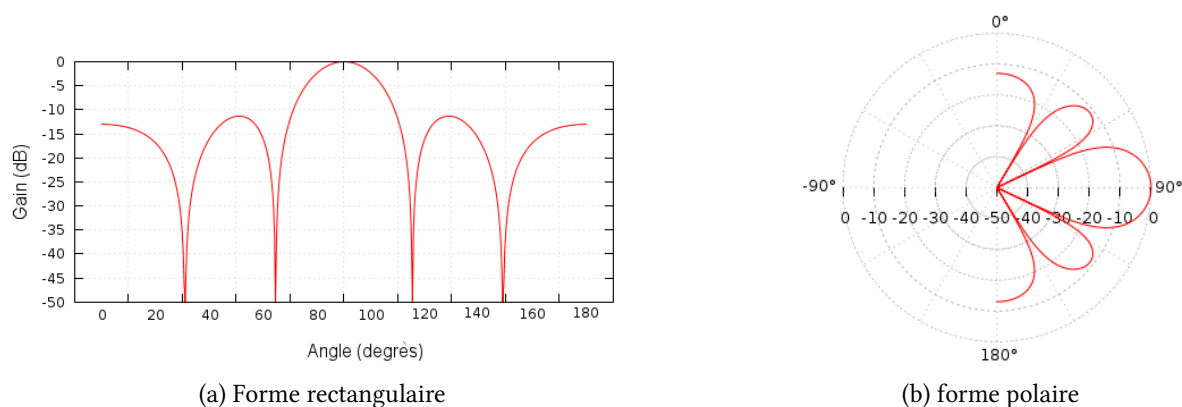


FIGURE 1.8 – Réponse spatiale, $N=4$, $d=0.2m$, $f=1kHz$.

1.2.4 L'espaceur entre microphones

Une des propriétés d'un réseau c'est que la largeur de faisceau diminue à mesure que l'espaceur d augmente. Donc, si nous voulons un faisceau plus précis, nous pouvons simplement diminuer l'espaceur d , mais on doit toujours éviter le recouvrement spatial (l'aliasing) causé par ce dernier. Le recouvrement spatial entraîne l'apparition de lobes parasites dans le diagramme de directivité, appelés lobes de grille, comme illustré dans la figure 1.9. Pour éviter cela dans la bande des fréquences qui nous intéresse, on doit avoir un déphasage entre deux microphones adjacents se trouvant dans l'intervalle $[-\pi, \pi]$. On peut calculer le déphasage de l'équation (1.6) qui est :

$$\phi_k = \frac{2\pi d \cos \theta_k}{\lambda} \quad (1.9)$$

on veut ϕ_k entre $-\pi$ et π , donc $d < \frac{\lambda_{min}}{2}$ où λ_{min} est le minimum des longueurs d'onde des signaux reçus sinon on va avoir plus d'un lobe principal. Par conséquent, la distance d'espaceur critique requise pour traiter les signaux dans la bande passante de la parole B_s (300-3400 Hz) est $d = 5 \text{ cm}$. Par analogie à la théorème de Nyquist, ce résultat peut être exprimé comme un théorème d'échantillonnage spatial.[6]

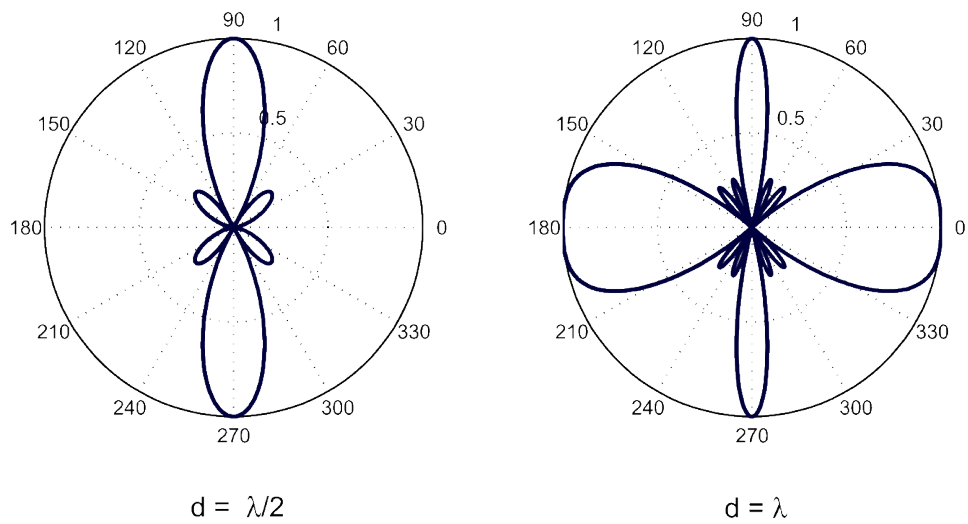


FIGURE 1.9 – Diagramme de directivité d'un réseau de microphones linéaire avec quatre éléments, en fonction de la direction horizontale θ , avec un échantillonnage spatial critique, $d = \lambda/2$ (gauche), et avec des effets d'alias pour λ (à droite).

Chapitre 2

Beamforming

Les signaux contiennent beaucoup d'informations sur les sources qui les produisent. Non seulement la forme d'onde de chaque signal exprime la nature de la source, ses caractéristiques temporelles et spatiales combinées aux lois de la physique nous permettent de déterminer l'emplacement de la source. Dans le monde réel, plusieurs sources en plus de celles utiles sont généralement présentes, et le bruit semble toujours contaminer les signaux mesurés. Ainsi, les méthodes de traitement du signal doivent se concentrer sur les signaux sélectionnés, ce qui nous permet de distinguer la cacophonie dans ses composants. Un exemple de focalisation est le filtrage linéaire : en appliquant des filtres passe-bande, les signaux qui occupent différentes bandes de fréquences peuvent être séparés. Le filtrage spatiotemporel doit être utilisé pour séparer les signaux en fonction de leurs directions de propagation et de leur contenu fréquentiel.

Ce chapitre décrit les opérations fondamentales qui produisent cette séparation.

Le filtrage spatial fait l'objet d'étude dans plusieurs domaines comme le radar, sonar, sismologie, communications, pour n'en citer que quelques-uns, Van Veen et Buckle [7] ont résumé plusieurs algorithmes de filtrage spatial, ces applications, ces avantages et inconvénients . Il peut être utilisé à plusieurs fins différentes à cause de sa flexibilité et ces capacités de filtrage. Dans une conférence ou un réunion, le signal de parole désiré provient d'un locuteur, et est corrompu par des signaux interférents tels que d'autres locuteurs et la réverbération de la pièce. Le filtrage spatial peut être utile dans un tel environnement, puisque les sources interférentes proviennent généralement de points dans l'espace, séparés du locuteur désiré. En exploitant la dimension spatiale du problème, le réseau de microphones tentent d'obtenir un signal vocal de haute qualité sans exiger que le locuteur parle directement dans un microphone. Lorsqu'un réseau est combiné avec des techniques de filtrage spatio-temporelles connues sous le nom de *beamforming*, on peut extraire l'information des signaux dont seul une mixture est observé. Cette opération peut être découplée en deux sous-processus : synchronisation et pondération-et-somme. La synchronisation se fait par retarder (ou avancer) la sortie de chaque capteur par un temps approprié afin que les signaux arrivant de la direction désirée soient en phase.

L'information qu'on doit avoir *a priori*, quelque soit la méthode utilisée, est la différence de temps

d'arrivée (TDOA), qui dépend de l'angle et la vitesse de propagation. Dans le filtrage spatial, on applique des poids sur les signaux et on ajoute les résultats pour n'avoir qu'une seule sortie. Cette étape est plus importante, car c'est là qu'on détermine les caractéristiques importantes (lobe principale et les lobes secondaires). Dans plusieurs applications les coefficients de pondération peuvent être déterminés *a priori*, mais c'est plus pratique de les estimer d'une manière adaptative dépendante du signal, la nature de bruit et les caractéristique de l'environnement.

Les formateurs de voies à base de filtre spatial ont été développés pour des signaux à bande étroite. Dans la plupart des applications, il y a deux hypothèses qui simplifient l'analyse : (I) l'hypothèse de la bande étroite, et (II) l'hypothèse du champ lointain. Pour de nombreuses applications de réseau de microphones, l'hypothèse (II) est valide. Cependant, l'hypothèse de la bande étroite n'est jamais valable. Pour la parole qui est à large bande, ces formateurs ne produisent pas la même réponse spatiale pour différentes fréquences, et la largeur du faisceau diminue à mesure que la fréquence augmente (voir la figure 2.1). Pour comprendre le problème inhérent à l'utilisation d'un réseau à bande étroite pour les signaux à large bande, considérons un réseau linéaire avec un nombre fixe d'éléments séparés par une distance inter-élément fixe. La dimension importante dans la mesure des performances du réseau est sa taille en termes de fréquence de fonctionnement. Ainsi, pour les signaux haute fréquence, un réseau fixe apparaîtra grand et le faisceau principal sera étroit. Cependant, pour les basses fréquences, le même réseau physique apparaît petit et le faisceau principal s'élargit.

Ceci est illustré dans la figure 2.1 qui montre la réponse d'un réseau conçu pour 1.5 kHz, mais opéré sur une gamme de fréquence de 300 Hz à 3 kHz. Donc si nous utilisons tels configurations, lorsque le sens de direction est différent de l'angle d'incidence de la source, la réponse serait un filtre passe-bas. En outre, le bruit ne sera pas atténué uniformément sur tout son spectre, ce qui entraînera des artefacts perturbateurs dans la sortie du réseau.

Par conséquent, il faut développer des techniques invariantes de formation de voies à large bande. L'une des méthodes est d'utiliser le formateur de voies à directivité constante (*constant directivity beamformer* ou CDB), qui est une classe spécifique de formateur de voies, conçu pour avoir la même réponse sur une large bande de fréquences.

La méthode la plus commune est de décomposer le spectre en plusieurs bandes, et d'utiliser un formateur de voies à bande étroite à chaque fréquence ; cette méthode est connu sous le nom de filtre-et-somme (filter-and-sum) [6] [8].

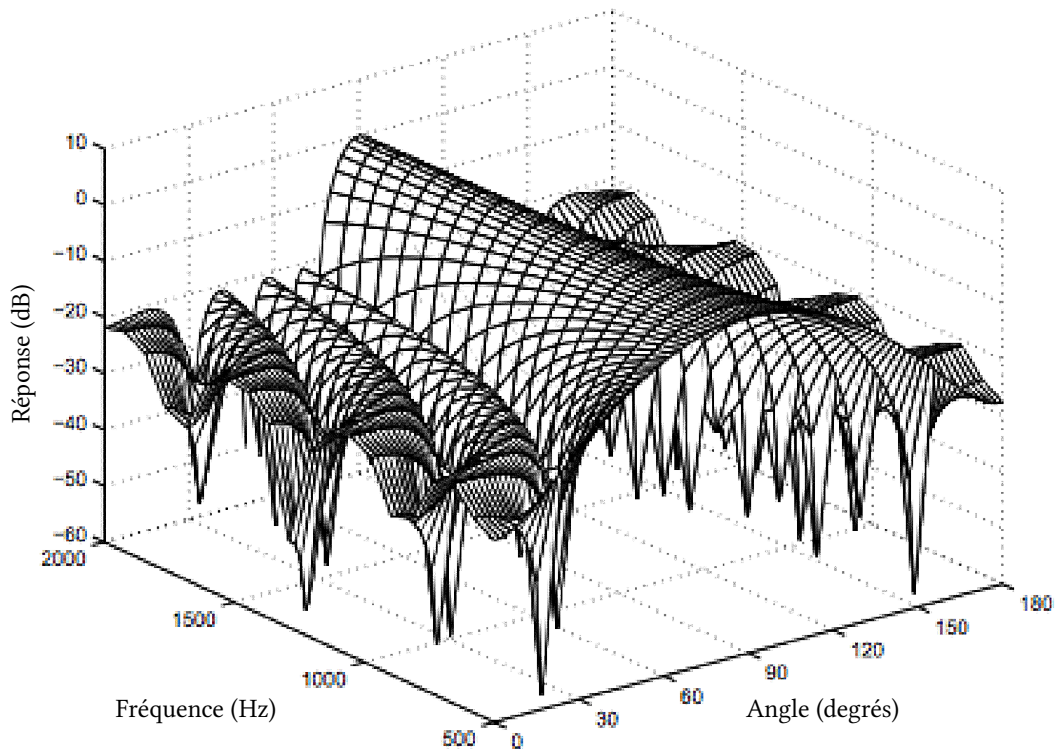


FIGURE 2.1 – réponse spatiale d’un réseau à bande étroite opérant dans une large bande.

2.1 Le filtre à retard numérique

Cette partie est très importante car il existe des formateurs de voies qui ne peuvent être utilisés que si le signal souhaité arrive au réseau de la direction directe, c’est-à-dire $\theta = 90^\circ$. Donc, avant de calculer la pondération, nous devons compenser le retard du signal reçu par le réseau de microphones via des filtres numériques afin de s’assurer que le signal désiré arrive au réseau de la direction normale au réseau. Le traitement de retard est appliqué au vecteurs d’échantillons du signal. Enfin, nous obtenons le lobe principal dirigé vers la direction souhaitée et les nuls vers les interférences.

La structure du filtre à retard numérique est représentée sur la figure 2.2. L’idée centrale est la compensation du retard τ_i dans le domaine temporel, ce qui rend l’angle d’incidence θ du signal 90° . La figure 2.2 montre que la structure du filtre à retard numérique est composé de deux parties, filtre à retard de temps entier et filtre à retard fractionnaire.

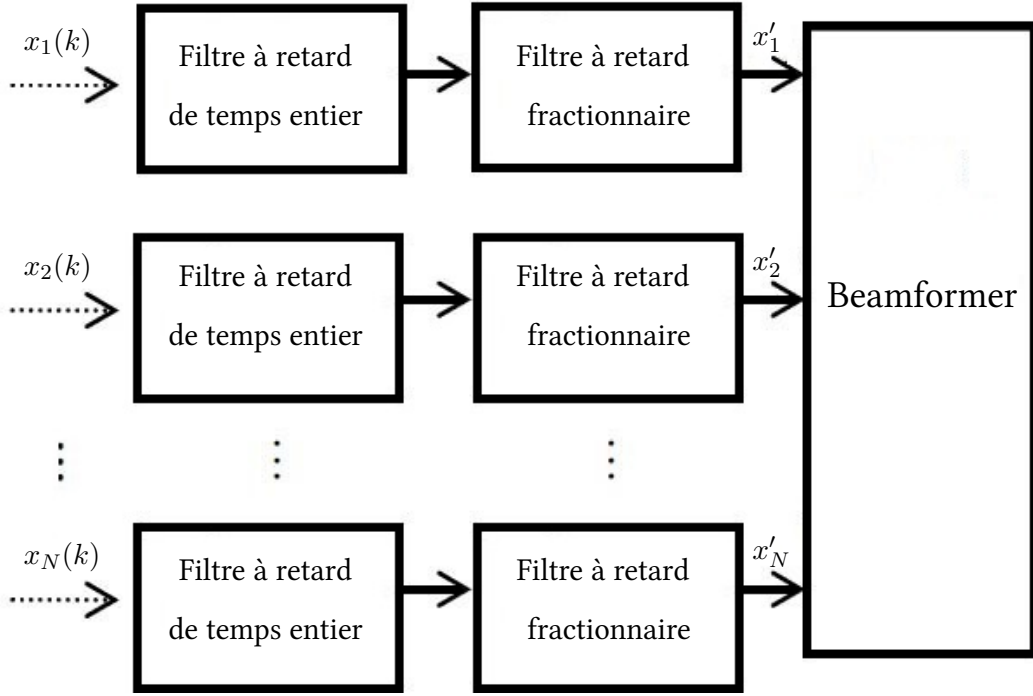


FIGURE 2.2 – La structure de filtre à retard numérique.

En supposant que la période d'échantillonnage est T_s , le délai à compenser dans le i ème élément de réseau peut être exprimé comme suit :

$$\frac{\tau_i}{T_s} = L + D$$

où le délai total peut être divisé en un délai de temps entier exprimé en nombre d'échantillons $L = \text{round}(\frac{\tau_i}{T_s})$, et un délai fractionnaire $D \in [0, 1[$, $D = \frac{\tau_i}{T_s} - L$.

Pour les données échantillonnées, le retard de temps entier, L , est d'abord effectué, puis le retard fractionnaire D créé par le filtre numérique. La compensation de L peut être obtenue par retard d'un multiple entier de l'intervalle d'échantillonnage T_s en utilisant une ligne de retard numérique, ayant une implémentation matérielle simple. La compensation de D est un peu compliquée, elle peut être réalisée en concevant des filtres FIR. En raison de sa faible valeur, l'ordre du filtre requis n'est pas élevé.

Il existe plusieurs techniques pour concevoir un filtre numérique $h(n)$ à retard fractionnaire, Laakso dans sa publication a présenté plusieurs techniques de conception de filtres à retard fractionnaire [9]. Dans notre cas on a utilisé l'interpolation de Lagrange. L'interpolation de Lagrange est l'une des plus simple techniques pour concevoir un filtre FIR d'approximation de délai fractionnaire, les coefficients sont obtenus par

$$h(n, D) = \prod_{k=0}^N \frac{D - k}{n - k} \quad n = 0, \dots, N. \quad (2.1)$$

comme le retard est très petit on n'en a besoin que d'un filtre d'ordre 2, un exemple détaillé est présenté dans la section suivante.

2.1.1 Exemple

Dans la figure 2.1.1, on a pris une portion de signal réel, de la parole, avec une fréquence d'échantillonnage de 8kHz. Les signaux des microphones d'un signal provenant de la direction de l'angle $\theta = 45^\circ$. On voit bien que les signaux sont décalés l'un par rapport à l'autre d'un décalage temporel fixe τ .

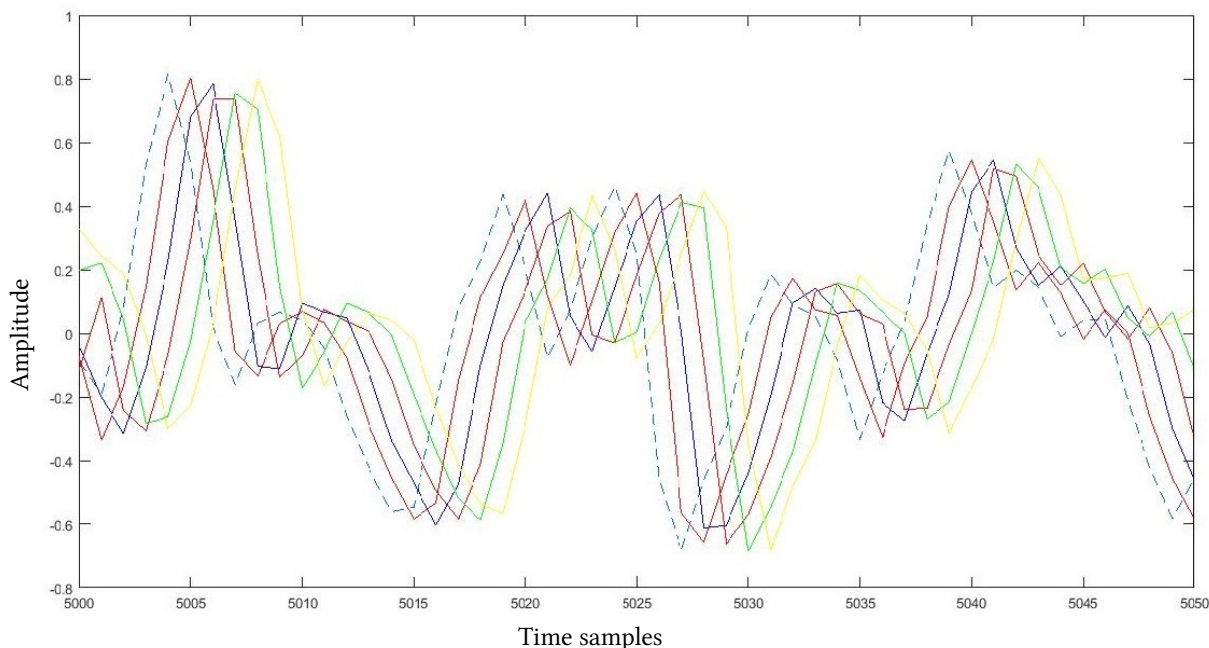


FIGURE 2.3 – Le signaux retardés d'une portion de signal échantillonné avec une fréquence de 8kHz.

Pour aligner les signaux on doit compenser l'avancement de certains signaux par un retard pour qu'ils soient en phase. On calcule le décalage entier nécessaire, qui est un nombre d'échantillons,

$$\tau_n = \frac{(i-1)d}{c} \cos \theta$$

Où $\theta = 45^\circ$ est l'angle du signal, $c = 340m/s$ la vitesse du son, $d = 5cm$ est la distance entre 2 microphones. Comme la fréquence d'échantillonnage est 8 kHz, la période entre 2 échantillons est $T = \frac{1}{8000} = 0.125ms$

On a $\tau_i = 0.1 \times (i-1) ms$ qui correspond à $D = 0.831 \times (i-1)$ échantillons. Pour le premier microphone (microphone de référence), $\tau_1 = 0ms$, donc on ne fait pas de décalage. Pour le 2ème, $D = 0.831$ échantillons qui est inférieur à un échantillon, en utilisant (2.1), et pour un décalage de $\tau_2 = 0.831$ échantillons, on détermine le filtre FIR $h(n, D)$ correspondant. Le nouveau signal retardé sera

$$x(k)_{retardé} = h(n, D) * x(k).$$

Où $*$ est la convolution.

Pour le 3ème microphone, on a $\tau_3 = 0.1 \times 2 = 0.2 \text{ ms}$ qui correspond à 1.662 échantillons, donc $D = 0.662 \text{ échantillons}$, d'abord on fait un décalage simple de tout le signal d'un échantillon, première colonne dans la figure 2.2, et après on refait le même traitement qu'on a fait avec le 2ème microphone.

Après le traitement avec tous les microphones on va obtenir tous les signaux alignés, comme montré dans la figure 2.1.1.

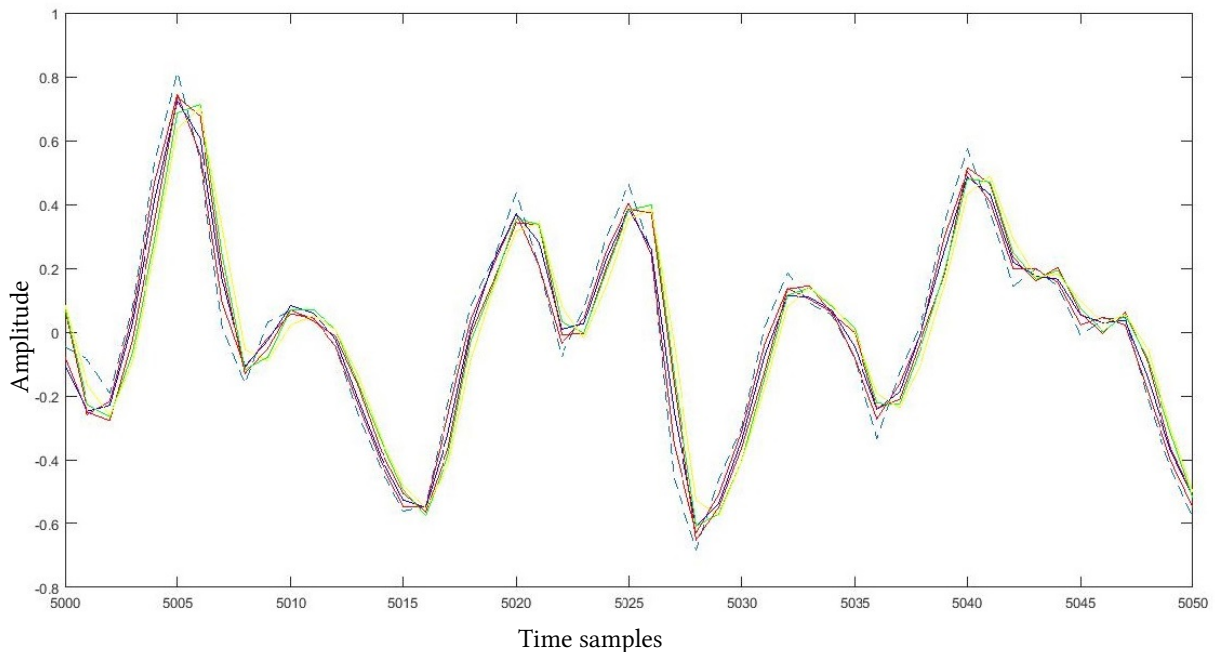


FIGURE 2.4 – Le signaux alignés.

2.2 retard et somme (Delay-and-sum)

Les avantages d'utiliser un réseau pour améliorer la réception du signal souhaité tout en supprimant simultanément le bruit et les interférences indésirable peuvent être illustrés par un formateur de voies à retard-et-somme. La formation des voies retard-et-somme, également connu sous le nom de "formation de voies classique", est l'une des techniques les plus simples et les plus anciennes pour la réalisation de systèmes de réseau directionnel. Cette technique est généralement utilisée en fonctionnement à bande étroite pour se concentrer sur un point ou une direction particulière. Un tel formateur se compose de deux étapes de traitement. La première étape consiste à déplacer temporairement chaque signal de capteur par une valeur correspondante à la TDOA entre ce capteur et la référence, à savoir τ_i . La deuxième étape consiste à additionner les signaux décalés dans le temps après les avoir pondéré.

Donc la sortie de cet formateur de voies est une somme pondérée des sorties d'éléments retardés :

$$y(t) = \sum_{i=1}^N w_i x_i(t - \tau_i) \quad (2.2)$$

où x_i est la sortie de $i^{\text{ème}}$ microphone, τ_i est le retard correspondant à ce dernier, et w_i et le facteur de pondération.

Les retards sont choisis pour compenser les différences dans les retards de propagation relatifs du point d'intérêt. Ainsi, les signaux provenant de l'emplacement désiré sont additionnés en phase, tandis que d'autres signaux subissent des interférences destructrices. En manipulant les poids et les retards, les lobes et les nuls de ce réseau peuvent être orientés dans les directions souhaitées. L'opération est montrée dans la figure 2.5.

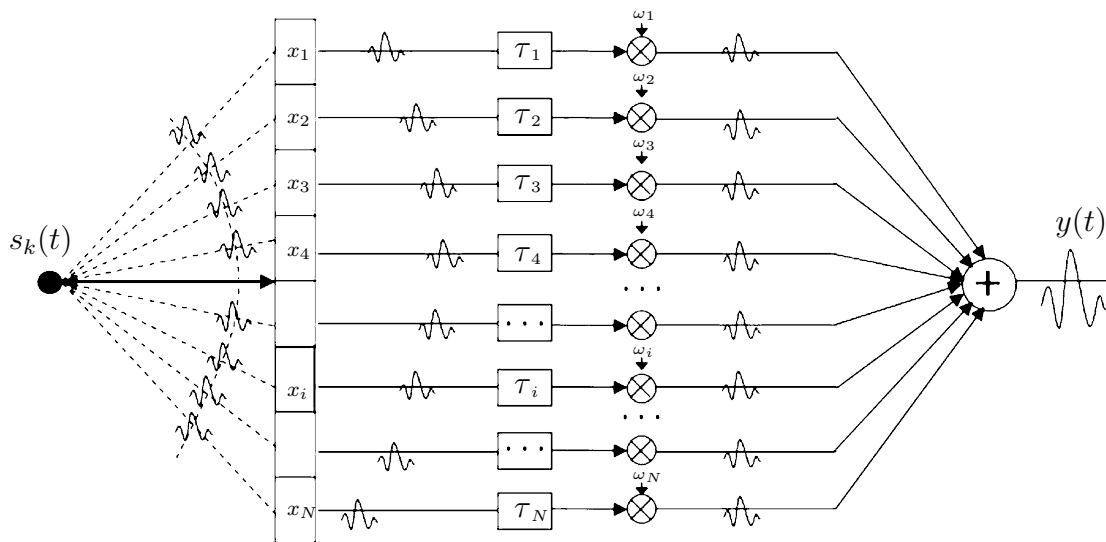


FIGURE 2.5 – Retard-et-somme.

Les facteurs de pondération w_i et les retards dans (2.2) sont calculés de telle manière à améliorer la forme du faisceau (avoir plus de directivité ce qui entraîne une meilleure résolution) et réduire les niveaux de lobes secondaires indésirables. Pour simplifier les choses, et pour avoir la même amplitude de signal en sortie, on choisit $w_1 = w_2 = w_n = 1/N$. Donc l'équation (2.2) devient :

$$y(t) = \frac{1}{N} \sum_{i=1}^N x_i(t - \tau_i). \quad (2.3)$$

Une des façons d'illustrer la performance de ce formateur de voies est d'examiner le diagramme de directivité correspondant, ou la réponse spatiale. Ce diagramme fournit une caractérisation complète du comportement du système dans une configuration spécifique, à savoir la fréquence et le dimensionnement

de réseau (le nombre de microphones et la distance inter-élément) étant donné que τ_i dépend à la fois de la géométrie du réseau et de la position de la source. Donc la réponse d'un delay-and-sum beamformer devrait être une fonction de f , d , N et de θ . À partir de l'équation (2.3), nous voyons qu'un Delay and sum beamformer est en effet un filtre spatial de N point et est défini par sa réponse directionnelle. Le $i^{\text{ème}}$ coefficient du filtre spatial est $\frac{1}{N}e^{j2\pi f\tau_i}$. La réponse directionnelle de ce filtre peut être trouvée en effectuant la transformée de Fourier.

Supposons maintenant que nous ayons un réseau linéaire uniforme comme celui de la figure 1.6, avec N microphone et une distance inter-éléments fixe d , et supposons aussi que la source est dans le champ lointain, et que la source arrive au réseau avec un angle θ , le TDOA entre le $i^{\text{ème}}$ microphone et celui de la référence peut être écrit comme :

$$\tau_i = (i - 1) \tau = (i - 1) \frac{d \cos \theta}{c}. \quad (2.4)$$

La transformée de Fourier du filtre DS peut être exprimée par :

$$\mathcal{S}_{DS}(\psi, \theta) = \frac{1}{N} \sum_{i=1}^N e^{-j \frac{2\pi f}{c} (i-1) d [\cos \psi - \cos \theta]}. \quad (2.5)$$

La réponse spatiale est définie par le module de la transformée de Fourier :

$$\mathcal{A}_{DS}(\psi, \theta) = \left| \frac{\sin[N\pi f d (\cos \psi - \cos \theta)/c]}{N \sin[\pi f d (\cos \psi - \cos \theta)/c]} \right|. \quad (2.6)$$

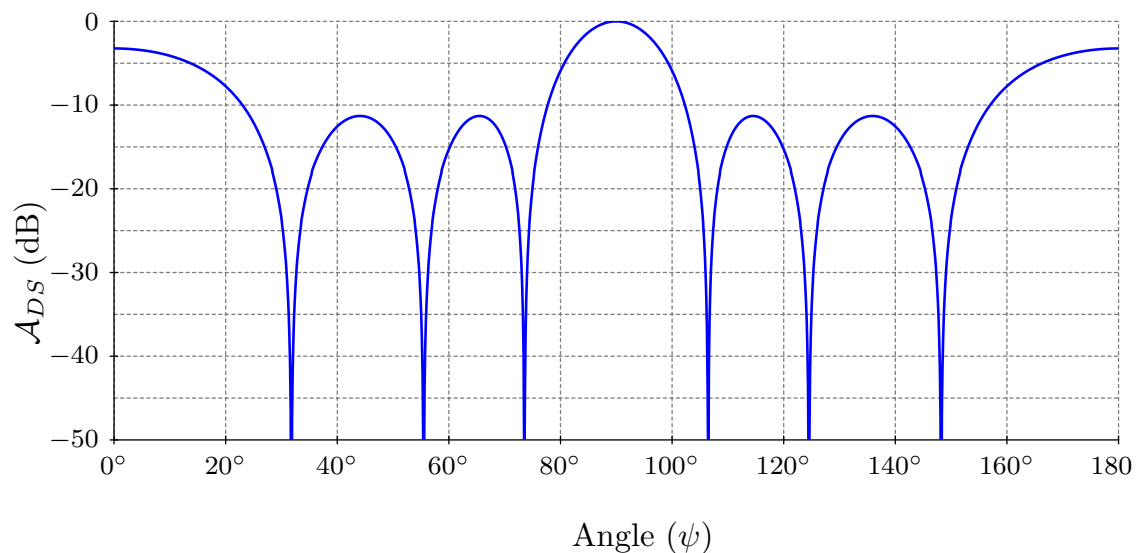


FIGURE 2.6 – Réponse d'un réseau de 10 microphones, $\theta = 90$, $d = 5$ cm, $f = 3.5$ kHz.

La figure 2.6 montre la réponse d'un réseau linéaire équidistant de dix capteurs avec $d = 5$ cm, $\theta = 90^\circ$ et $f = 3.5$ kHz. Cette réponse se compose d'un total de 9 voies (en général, le nombre de voies compris entre 0 et 180° est égal à $N-1$). Celui avec l'amplitude la plus élevée est le lobe principal (mainlobe) et les autres s'appellent les lobes secondaires. Un paramètre important concernant le lobe principal est sa largeur, qui est définie comme la région entre les premiers zéros de chaque côté. Avec un réseau linéaire uniforme, la largeur du lobe principal peut être trouvée en cherchant les zéros de \mathcal{A}_{DS} , on la trouve égale à $2 \cos^{-1}[c/(Ndf)]$. Cette largeur diminue avec l'augmentation du nombre de microphones, l'espacement entre les microphones voisins et la fréquence du signal [6].

Les lobes secondaires représentent le gain de bruit et des interférences présentes dans des directions autres que la direction désirée. Dans la conception d'un réseau, nous voulons que les lobes secondaires soient aussi bas afin que les signaux indésirables soient atténués autant que possible. De plus, avec un filtre spatial de longueur N , il existe toujours $N - 1$ nuls. Nous pouvons concevoir les coefficients de pondération afin que ces nuls soient placés selon les directions des sources concurrentes. Ceci est lié à des techniques de la formation de voies adaptatif qui sont traitées dans la section 2.5.

2.3 Minimum de Variance Réponse Non Distordue (MVDR)

Le principe de cette méthode de formation de voies consiste à choisir les poids ω_i minimisant le bruit et les interférences arrivant de directions différentes de θ à la sortie, tout en gardant un gain fixe dans la direction θ . Ceci est équivalent à la maximisation du rapport signal sur interférences plus bruit (SINR) en sortie du formateur de voies. Le MVDR est un formateur de voies dépendant du signal, c'est-à-dire que les coefficients du filtre dépendent des propriétés statistiques des signaux reçus. Les puissances des signaux utiles et d'interférence sont estimées à partir des matrices de corrélation des x_i et les poids sont calculés de façon à maximiser le rapport de ces puissances et annuler éventuellement le bruit. En présence d'interférences, le signal reçu au niveau du réseau de microphones s'écrit :

$$\mathbf{x}(t) = \mathbf{a}(\theta_0)s(t) + \mathbf{x}_I(t) + \mathbf{n}(t) \quad (2.7)$$

où \mathbf{x}_I représente le vecteur des signaux interférents.

L'expression du signal en sortie est donnée par :

$$y(t) = \mathbf{w}^H \mathbf{a}(\theta_0) s(t) + \mathbf{w}^H \mathbf{n}_I(t). \quad (2.8)$$

avec $\mathbf{n}_I(t)$ la somme de bruit plus interférences, $(\cdot)^H$ est l'opérateur Hermitien.

La puissance du signal $s(t)$ à la sortie de formateur, P_{signal} s'écrit :

$$P_{signal} = E \left[\{ \mathbf{w}^H \mathbf{a}(\theta_0) s(t) \} \{ \mathbf{w}^H \mathbf{a}(\theta_0) s(t) \}^H \right]. \quad (2.9)$$

soit

$$P_{signal} = E \left[\left| \mathbf{w}^H \mathbf{a}(\theta_0) s(t) \right|^2 \right]. \quad (2.10)$$

En notant P_s la puissance du signal $s(t)$: $P_s = E[s(t)s(t)^H]$

$$P_{signal} = P_s \left| \mathbf{w}^H \mathbf{a}(\theta_0) \right|^2 \quad (2.11)$$

Et la puissance des bruits et interférences en sortie s'écrit :

$$P_{n_I} = E \left[\left\{ \mathbf{w}^H \mathbf{n}_I(t) \right\} \left\{ \mathbf{w}^H \mathbf{n}_I(t) \right\}^H \right]. \quad (2.12)$$

soit :

$$\begin{aligned} P_{n_I} &= \mathbf{w}^H E \left[\mathbf{n}_I(t) \mathbf{n}_I(t)^H \right] \mathbf{w} \\ &= \mathbf{w}^H \mathbf{R}_{n_I n_I} \mathbf{w}. \end{aligned} \quad (2.13)$$

Avec $\mathbf{R}_{n_I n_I}$ la matrice de covariance de bruit plus interférences.

Le SINR représente le rapport de la puissance du signal utile à celle des interférences et du bruit. D'après les équations 2.12 et 2.13, on obtient :

$$SINR = \frac{P_s \left| \mathbf{w}^H \mathbf{a}(\theta_0) \right|^2}{\mathbf{w}^H \hat{\mathbf{R}}_{n_I n_I} \mathbf{w}}. \quad (2.14)$$

La maximisation du SINR est équivalente à la minimisation de la puissance du bruit et des interférences en sortie de ce formateur de voies, qui n'est pas connue *a priori*, donc on la remplace avec la puissance totale. Calculer la pondération MVDR revient à résoudre :

$$\min_{\mathbf{w}} \mathbf{w}^H \hat{\mathbf{R}}_{xx} \mathbf{w}. \quad (2.15)$$

sous la contrainte

$$\mathbf{w}^H \mathbf{a}(\theta_0) = 1 \quad (2.16)$$

D'où vient son nom MV (2.15) et DR (2.16). Ce problème peut être résolu en utilisant les multiplicateurs de Lagrange. La solution obtenue est donnée par :

$$\mathbf{w}_{mvd} = \frac{\hat{\mathbf{R}}_{xx}^{-1} \mathbf{a}(\theta_0)}{\mathbf{a}^H(\theta_0) \hat{\mathbf{R}}_{xx}^{-1} \mathbf{a}(\theta_0)}. \quad (2.17)$$

Bien que la méthode MVDR soit robuste et fournisse de bons résultats, elle est sensible aux erreurs sur la direction du signal utile qui peuvent causer une dégradation significative des performances du système.

2.3.1 MVDR fréquentiel

Dans cette partie, nous développons l'implémentation du formateur de voies MVDR dans le domaine fréquentiel (figure 2.7). Nous prenons la transformée de Fourier discrète de $x(t)$ sur l'intervalle de temps $(k-1)\Delta T \leq t < k\Delta T$ à un ensemble de fréquences f_m , qui sont séparées par $\Delta_f = 1/\Delta T$. Nous supposons que ΔT est assez grand afin que les échantillons soient statistiquement indépendants. Après, nous implémentons le formateur de voies à bande étroite expliqué récemment à chacune des M fréquences.

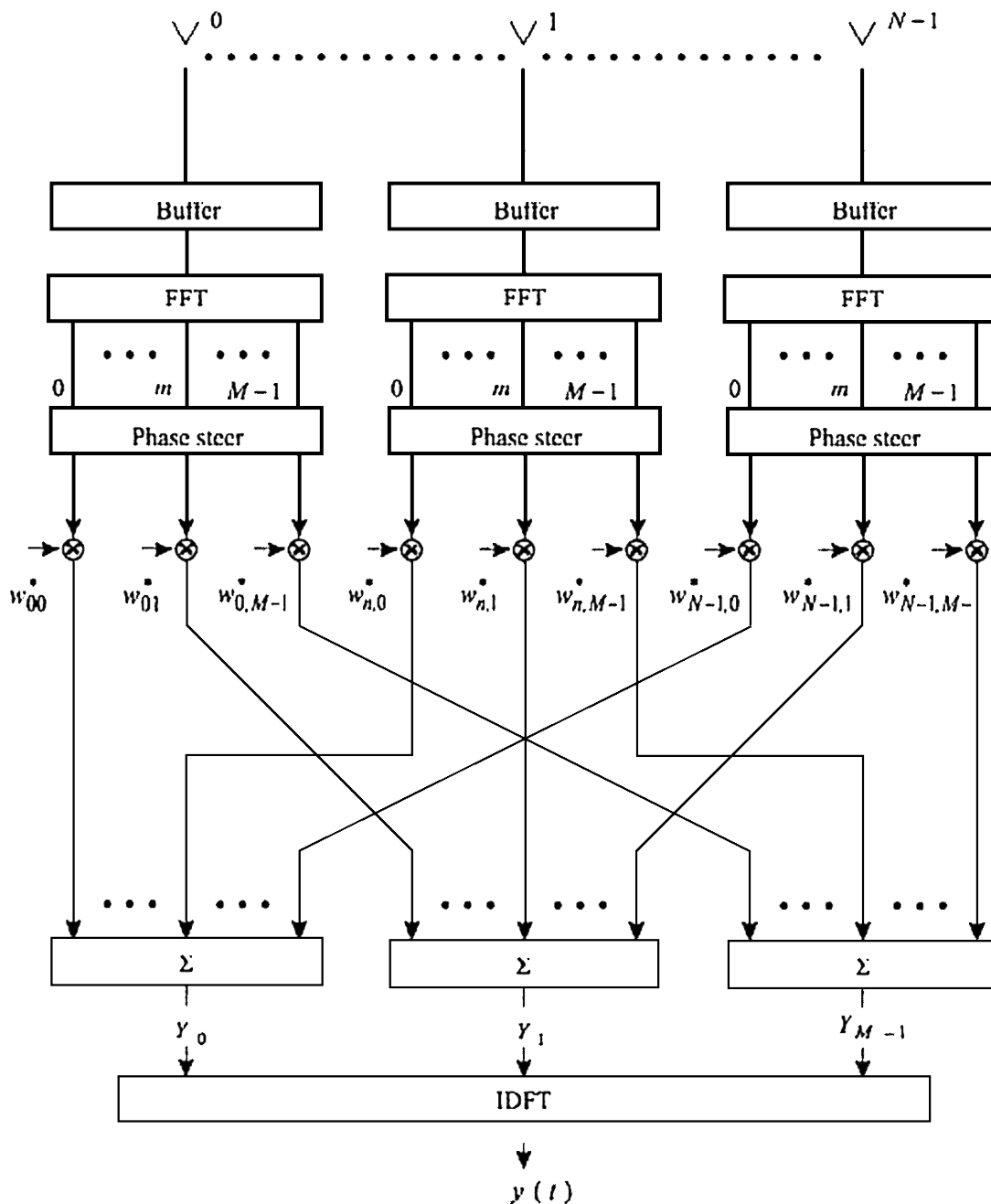


FIGURE 2.7 – Le bloc de traitement de MVDR fréquentiel.

En pratique, nous implémentons le filtre spatial fréquentiel en échantillonnant $x(t)$ chaque $1/2f_{max}$ secondes, où f_{max} est la fréquence maximale de signal, et nous obtenons M échantillons. Nous effectuons alors TFD (Transformée de Fourier Discrète) pour obtenir les M échantillons de fréquence. Nous accomplissons l'opération TFD Avec une FFT(Transformée de Fourier Rapide) à cause de ça facilité d'implémentation. Le concept est simple, mais il existe des détails qui doivent être développés pour effectuer la mise en œuvre [10] [11].

On va diviser le traitement sur plusieurs étapes :

Étape 1 : générer les vecteurs dans le domaine fréquentielle

Un réseau de microphones avec le traitement TFD est illustré à la figure 2.7. La sortie de chaque capteur est échantillonnée chaque T_s seconds où

$$T_s = \frac{1}{f_s}$$

avec f_s la fréquence d'échantillonnage qui doit respecter la théorème de Shanon, pour la bande de la parole il suffit que $f_s > 2f_{max} = 7$ kHz. Nous prenons la TFD de M échantillons. L'intervalle de L'observation totale sur chaque M échantillons est

$$\Delta T = MT_s.$$

La sortie du TFD est un ensemble de M valeurs de fréquences qui sont séparées en fréquence de $1/MT_s$. Ainsi, la résolution en fréquence est

$$\Delta f = 1/MT_s = f_s/M$$

Plus M est grand, plus on a de la résolution, et le choix de M dépend du besoin de l'algorithme.

On note t_0 l'instant d'arrivé d'échantillon le plus récent. L'échantillon actuel du n ème capteur est $x_n(t_0)$. L'échantillon précédent est $x_n(t - T_s)$. Nous introduisons la notation

$$\begin{aligned} x_n(t_0) &= x_n(0) \\ x_n(t_0 - (M - 1)T_s) &= x_n(M - 1) \end{aligned} \quad (2.18)$$

On définit le vecteur des échantillons temporels $\tilde{\mathbf{x}}_n$,

$$\tilde{\mathbf{x}}_n = \begin{bmatrix} x_n(t_0 - (M - 1)T_s) \\ \vdots \\ x_n(t_0) \end{bmatrix} = \begin{bmatrix} x_n(M - 1) \\ \vdots \\ x_n(0) \end{bmatrix}, \quad n = 0, \dots, N - 1. \quad (2.19)$$

La transformée de fourrier de $\tilde{\mathbf{x}}_n$:

$$\tilde{\mathbf{X}}_n = \left[X_n\{0\} \quad \dots \quad X_n\{m\} \quad \dots \quad X_n\{M - 1\} \right]^T. \quad (2.20)$$

Pour chaque fréquence m on construit un vecteur $\mathbf{X}\{m\}$ à partir de tous les N microphones du réseau

$$\mathbf{X}\{m\} = \begin{bmatrix} X_0\{m\} & X_1\{m\} & \cdots & X_{N-1}\{m\} \end{bmatrix}^T, m = 0, \dots, M-1. \quad (2.21)$$

L'argument m représente le numéro du bin de fréquence.

Nous traitons chaque bin de fréquence de manière indépendante en utilisant l'algorithme MVDR.

Étape 2 : le vecteur de direction du réseau

Pour construire la pondération pour la m ème fréquence, nous avons besoin du vecteur de direction et de la matrice spectrale spatiale. Le vecteur de direction du réseau en fonction de la fréquence est

$$\mathbf{a}_m = \begin{bmatrix} e^{-j2\pi f_m \tau_0} & e^{-j2\pi f_m \tau_1} & \dots & e^{-j2\pi f_m \tau_{N-1}} \end{bmatrix}^T. \quad (2.22)$$

Où τ_N est le retard du i ème microphone par rapport à l'origine des coordonnées, et qui dépend seulement de θ et la distance d . L'indice m indique que nous conservons explicitement la dépendance en fréquence.

La fréquence dans la m ème bin est

$$f_m = m\Delta f, \quad m = 0, \dots, M-1.$$

Où

$$\Delta f = \frac{1}{MT_s} = \frac{f_s}{M}.$$

pour notre cas de réseau linéaire nous avons

$$\tau_i = \frac{(i-1) \cos \theta}{c}. \quad (2.23)$$

Étape 3 : La matrice spectrale spatiale

La matrice spatiale spectrale dans la m ème fréquence est

$$\mathbf{R}_{\mathbf{xx}}\{m\} = E \left[\mathbf{X}\{m\} \mathbf{X}^H\{m\} \right]. \quad (2.24)$$

on estime $\mathbf{R}_{\mathbf{xx}}\{m\}$ à partir des échantillons de façon adaptatif. On mets les dernières F FFTs dans un buffer et on les utilise pour estimer $\hat{\mathbf{R}}_{\mathbf{xx}}\{m\}$. Donc

$$\hat{\mathbf{R}}_{\mathbf{xx}}\{m\} = \frac{1}{F} \sum_{i=0}^{F-1} \mathbf{X}_i\{m\} \mathbf{X}_i^H\{m\}. \quad (2.25)$$

F est choisi d'une façon à avoir un compromis entre la résolution et la rapidité de calcul.

Étape 4 : le traitement en bande étroite

Nous traitons chaque bin de fréquence en utilisant les algorithmes utilisés en bande étroite. Le formateur de voies MVDR est utilisé.

$$\mathbf{W}_{mvd} \{m\} = \frac{\hat{\mathbf{R}}_{\mathbf{x}\mathbf{x}}^{-1} \{m\} \mathbf{a}_m^H(\theta)}{\mathbf{a}_m^H(\theta) \hat{\mathbf{R}}_{\mathbf{x}\mathbf{x}}^{-1} \{m\} \mathbf{a}_m(\theta)}, \quad m = 0, \dots, M - 1. \quad (2.26)$$

Après la pondération on somme les composantes de la même fréquence. On définit le vecteur de sortie dans le domaine fréquentiel comme

$$\tilde{\mathbf{Y}} = \left[Y\{0\} \quad \dots \quad Y\{m\} \quad \dots \quad Y\{M - 1\} \right]^T. \quad (2.27)$$

L'étape finale consiste à prendre la transformée de fourrier inverse (TFDI) pour obtenir M échantillons de sortie dans le domaine temporel.

$$\tilde{\mathbf{y}} = \left[y\{M - 1\} \quad y\{M - 2\} \quad \dots \quad y\{0\} \right]^T. \quad (2.28)$$

2.4 Formateurs de voies à filtre à réponse impulsionnelle finie (FIR)

Nous choisissons la réponse impulsionnelle (ou les poids) pour obtenir la réponse fréquentielle souhaitée, (figure 2.8). En pratique, ces poids sont déterminés de manière adaptative à partir des données. Un grand avantage de l'implémentation temporelle c'est qu'elle nous permet d'actualiser le formateur de voies chaque fois qu'un nouveau échantillon arrive, contrairement à l'implémentation fréquentielle qui nécessite un bloc de données pour effectuer la FFT, ce qui la rend plus efficace et flexible avec le temps.

2.4.1 L'idée de base de cet algorithme

L'algorithme est capable de maintenir une réponse de fréquence choisie dans la direction d'observation tout en minimisant le bruit de sortie à cause d'une relation simple entre la réponse fréquentielle de la direction d'observation et les poids w_i . Supposons que la direction d'observation est choisie perpendiculaire à la ligne des capteurs. Donc, des signaux identiques arrivent sous forme d'une onde plane à la ligne des microphones. Cependant, la forme d'onde du bruit arrivant d'une direction autre que la direction d'observation sont généralement différents d'un capteur à un autre. Les signaux (signal plus le bruit) à chaque colonne sont multipliées par les poids et ajoutées pour former la sortie du réseau. Ainsi, en ce qui concerne le signal désiré, le processus est équivalent à une seule ligne à retard temporel dans laquelle chaque poids est égal à la somme des poids dans la colonne verticale. Ces poids doivent être sélectionnés

de manière à donner la caractéristique de réponse fréquentielle souhaitée dans la direction d'observation, c'est-à-dire de garder le signal désiré tel qu'il est et annuler les interférences et le bruit.

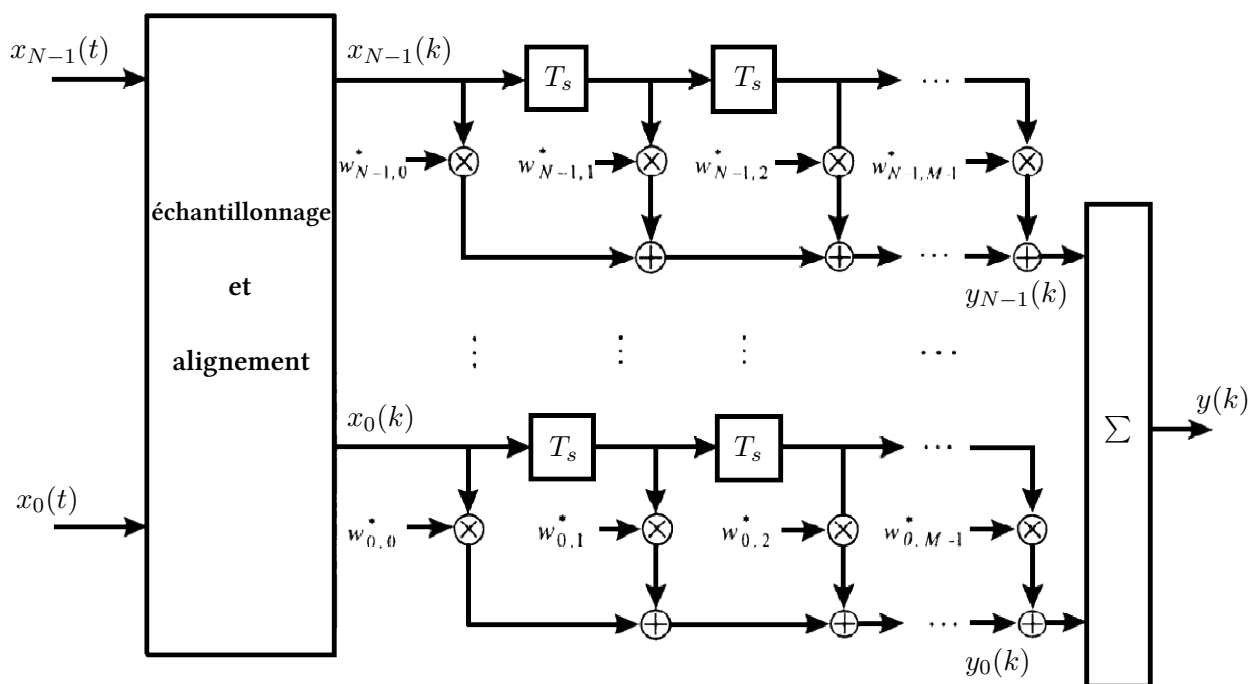


FIGURE 2.8 – Bloc de traitement de filtre à réponse impulsionnelle finie.

Si la direction d'observation est choisie autrement que perpendiculaire à la ligne de capteurs, le réseau peut être dirigé mécaniquement ou électriquement en ajoutant des retards de temps (comme représenté dans la partie alignement de la figure 2.8) placés après chaque capteur.

Un processus ayant N microphones et M colonne par microphone a NM poids et nécessite M contraintes pour déterminer sa réponse fréquentielle. Les $(N - 1)M$ degrés de liberté restants dans le choix des poids peuvent être utilisés pour minimiser la puissance de bruit. Puisque la réponse en fréquence de la direction d'observation est fixée par les M contraintes, la minimisation de la puissance de bruit des directions non désirées équivaut à minimiser la puissance de totale de la sortie du réseau. Ceci est possible tant que le signal désiré n'est pas corrélé avec le bruit.

2.4.2 Le modèle FIR

le modèle du FIR qui nous intéresse est illustré dans la figure 2.8. Pour simplifier les calculs, on considère que le réseau est positionner sur la direction d'observation en utilisant la méthode "Le filtre à retard numérique" expliqué précédemment (Digital delay).

L'échantillon rentrant au premier étage (les poids $w_{0,0}, \dots, w_{N-1,0}$ de la figure 2.8) est

$$\mathbf{x}_0(t_0) = \begin{bmatrix} x_0(t_0) \\ \vdots \\ x_{N-1}(t_0) \end{bmatrix} = \begin{bmatrix} x_{00} \\ \vdots \\ x_{N-1,0} \end{bmatrix} = \mathbf{x}_0. \quad (2.29)$$

De la même façon pour la m ème colonne de poids

$$\mathbf{x}_m(t_0) = \begin{bmatrix} x_0(t_0 - mT_s) \\ \vdots \\ x_{N-1}(t_0 - mT_s) \end{bmatrix} = \begin{bmatrix} x_{0,m} \\ \vdots \\ x_{N-1,m} \end{bmatrix} = \mathbf{x}_m. \quad m = 0, \dots, M-1. \quad (2.30)$$

On peut concaténer les vecteurs $\mathbf{x}_0, \dots, \mathbf{x}_m$ pour former un seul vecteur de taille $NM \times 1$

$$\mathbf{x} = \begin{bmatrix} \mathbf{x}_0 \\ \vdots \\ \mathbf{x}_{M-1} \end{bmatrix}. \quad (2.31)$$

On définit le vecteur directionnel dans la m ème colonne

$$\mathbf{a}_m(f, \theta) = \left[e^{-j2\pi f(\tau_0 + mT_s)} \quad e^{-j2\pi f(\tau_1 + mT_s)} \quad \dots \quad e^{-j2\pi f(\tau_{N-1} + mT_s)} \right]^T. \quad (2.32)$$

Les poids correspondent aux m ème colonne du bloc FIR est

$$\mathbf{w}_m = \begin{bmatrix} \omega_{0m} \\ \vdots \\ \omega_{N-1m} \end{bmatrix}. \quad (2.33)$$

On concatène ces vecteurs pour construire un vecteur de poids de taille $NM \times 1$,

$$\mathbf{w} = \begin{bmatrix} \mathbf{w}_0 \\ \vdots \\ \mathbf{w}_{M-1} \end{bmatrix}. \quad (2.34)$$

Donc la sortie de l'instant t_0 sera

$$y(t_0) = \sum_{m=0}^{M-1} \mathbf{w}_m^H \mathbf{x}_m. \quad (2.35)$$

Ou en d'autre terme

$$y(t_0) = \mathbf{w}^H \mathbf{x} \quad (2.36)$$

2.4.3 Le formateur de voies de Frost

L'algorithme de Frost est une ancienne technique de formation de voies (1970) qui réduit le bruit tout en maintenant le signal de direction désirée. C'est un système adaptatif qui minimise l'énergie du bruit en ajustant les poids des filtres. Cet algorithme peut être utile non seulement dans les réseaux de microphones pour l'amélioration de la parole, mais aussi dans les communications, le radar et le sonar. Il existe différentes façons de définir les contraintes qui sont intrinsèquement construites dans la structure de cet algorithme [12] [13].

On introduit la contrainte dans la direction d'observation, on définit un vecteur de taille $NM \times 1$, $\mathbf{c}_m, m = 0, \dots, M - 1$. comme

$$\mathbf{c}_m = \begin{bmatrix} \mathbf{0}_N \\ \mathbf{0}_N \\ \vdots \\ \mathbf{1}_N \\ \vdots \\ \mathbf{0}_N \end{bmatrix}. \quad (2.37)$$

où $\mathbf{1}_N$ est un vecteur de taille $N \times 1$ situé dans la m ème position.

Après, on définit la matrice \mathbf{C} , de taille $NM \times M$

$$\begin{aligned} \mathbf{C} &= \begin{bmatrix} \mathbf{c}_0 & \mathbf{c}_1 & \cdots & \mathbf{c}_{M-1} \end{bmatrix} \\ &= \begin{bmatrix} \mathbf{1}_N & \mathbf{0} & \cdots & \mathbf{0} \\ \mathbf{0} & \mathbf{1}_N & \cdots & \mathbf{0} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{0} & \mathbf{0} & \cdots & \mathbf{1}_N \end{bmatrix}. \end{aligned} \quad (2.38)$$

L'équation de contrainte est

$$\mathbf{w}^H \mathbf{C} = \mathbf{g}^H \quad (2.39)$$

Où

$$\mathbf{g} = \begin{bmatrix} g_0 & g_1 & \cdots & g_{M-1} \end{bmatrix}^T. \quad (2.40)$$

Les valeurs de \mathbf{g} déterminent la réponse spatiale du formateur de voies, chaque g_m représente la réponse pour une certaine direction θ_m .

Le formateur de voies de Frost impose que la réponse soit non-distordue dans la direction d'observation, donc

$$\mathbf{g}^H = \begin{bmatrix} 1 & 0 & \cdots & 0 \end{bmatrix}. \quad (2.41)$$

Le problème d'optimisation est le suivant :

On minimise la fonction de coût qui est défini par la variance du signal de sortie

$$P_0 = \mathbf{w}^H \mathbf{R}_{xx} \mathbf{w}. \quad (2.42)$$

Sous la contrainte

$$\mathbf{w}^H \mathbf{C} = \mathbf{g}^H. \quad (2.43)$$

Où \mathbf{R}_{xx} est la matrice de covariance.

Ce problème d'optimisation peut être résolu, comme dans la méthode de MVDR, en utilisant le multiplicateur de Lagrange λ de taille $M \times 1$, la seule chose qui diffère c'est la contrainte.

Donc on définit la fonction de cout de Lagrange

$$\mathcal{L}(\mathbf{w}, \lambda) = \frac{1}{2} \mathbf{w}^H \mathbf{R}_{xx} \mathbf{w} + \lambda^H (\mathbf{C}^H \mathbf{w} - \mathbf{g}). \quad (2.44)$$

On prend le gradient de \mathcal{L} en respectant \mathbf{w} , et on la met égale à 0,

$$\frac{\partial \mathcal{L}(\mathbf{w}, \lambda)}{\partial \mathbf{w}} = \mathbf{w}^H \mathbf{R}_{xx} + \lambda^H \mathbf{C}^H = 0. \quad (2.45)$$

donc

$$\mathbf{w}^H = -\lambda^H \mathbf{C}^H \mathbf{R}_{xx}^{-1}. \quad (2.46)$$

En utilisant la contrainte

$$-\lambda^H \mathbf{C}^H \mathbf{R}_{xx}^{-1} \mathbf{C} = \mathbf{g}^H. \quad (2.47)$$

résoudre pour λ^H et met dans (2.46) donne le résultat final,

$$\mathbf{w}_{frost}^H = \mathbf{g}^H \left[\mathbf{C}^H \mathbf{R}_{xx}^{-1} \mathbf{C} \right]^{-1} \mathbf{C}^H \mathbf{R}_{xx}^{-1}. \quad (2.48)$$

A noter que la taille de \mathbf{w}_{frost}^H est $NM \times 1$. Ce résultat est connu sous le nom de l'algorithme de Frost. Plus tard, on va discuter la version adaptative de cet algorithme.

2.4.4 Le formateur de voies Linearly Constrained Minimum Variance (LCMV)

Dans la formateur de voies LCMV, les coefficients de filtre sont ajustés en fonction des statistiques des signaux de sortie. Pour décrire ces statistiques, nous utilisons l'opérateur de l'espérance dont l'argument est un processus stochastique.

L'extension de l'algorithme de Frost à l'algorithme LCMV est un peu simple. Au lieu d'utiliser une seule colonne de contrainte, on utilise une matrice qui contient N_c colonnes [10]. Donc pour N_c contraintes on définit la matrice \mathbf{c}_m de taille $NM \times N_c$

$$\mathbf{c}_m = \begin{bmatrix} \mathbf{0}_{NN_c} \\ \mathbf{0}_{NN_c} \\ \vdots \\ \mathbf{c}_{N_c} \\ \vdots \\ \mathbf{0}_{NN_c} \end{bmatrix}, \quad (2.49)$$

Où \mathbf{c}_{N_c} est une matrice de taille $N \times N_c$, située dans la m ème position.

Puis \mathbf{C} est une matrice de contraintes de taille $NM \times N_cM$

$$\mathbf{C} = \begin{bmatrix} \mathbf{c}_{N_c} & \mathbf{0}_{NN_c} & \cdots & \mathbf{0}_{NN_c} \\ \mathbf{0}_{NN_c} & \mathbf{c}_{N_c} & & \mathbf{0}_{NN_c} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{0}_{NN_c} & \mathbf{0}_{NN_c} & \cdots & \mathbf{c}_{N_c} \end{bmatrix}. \quad (2.50)$$

\mathbf{g} une matrice de $N_cM \times 1$

$$\mathbf{g}^H = \begin{bmatrix} \mathbf{g}_1^H & \mathbf{0}_{N_c}^H & \cdots & \mathbf{0}_{N_c}^H \end{bmatrix}. \quad (2.51)$$

Où \mathbf{g}_1^H est la même défini en (2.40) et en (2.41) mais avec N_c colonnes.

En utilisant la même méthode d'optimisation précédente, le vecteur de poids est

$$\mathbf{w}_{lcmv}^H = \mathbf{g}^H \left[\mathbf{C}^H \mathbf{R}_x^{-1} \mathbf{C} \right]^{-1} \mathbf{C}^H \mathbf{R}_x^{-1} \quad (2.52)$$

Ce résultat est identique à celui de Frost, mais avec \mathbf{C} et \mathbf{g} différents. Le calcul de (??) implique l'inversion de \mathbf{R}_{xx} . En pratique, les estimations pour \mathbf{R}_{xx} peuvent être mal conditionnées, en particulier pour les grandes longueurs de filtre et les signaux d'entrée colorés (tels que la parole). De plus, inverser \mathbf{R}_{xx} demande beaucoup de calcul. Une solution large bande peut être obtenue en transformant le problème de minimisation avec contrainte en une sans contrainte. C'est le principe de Generalized sidelobe canceller (GSC).

2.4.5 Annuller des lobes secondaires généralisé (Generalized sidelobe canceller)

Griffith et Jim ont introduit le Generalized Sidelobe Canceller (GSC) comme une formulation alternative du problème LCMV qui transforme le problème de minimisation à contrainte (2.43) en un sans contrainte [14].

La structure GSC est une structure de formation de voies qui peut être utilisée pour implémenter une grande variété de formateurs de voies adaptatifs à contraintes linéaires. La structure est conçue pour être utilisée avec des réseaux qui ont été orientés de manière temporelle de telle sorte que le signal utile apparait approximativement en phase aux sorties dirigées. Un avantage majeur de la nouvelle structure est que les contraintes peuvent être implémentées à l'aide d'un simple matériel. La structure montre que l'on incorpore des algorithmes qui ont été suggérés précédemment pour être utilisés dans des méthodes adaptatives ainsi que pour inclure de nouvelles approches.

Le GSC est basé sur une décomposition du vecteur \mathbf{w} en deux composants orthogonaux \mathbf{w}_c et \mathbf{w}_b . Ces deux composants distincts sont représentés par deux blocs de traitement.

Le premier bloc qui est un formateur de voies conventionnel se compose d'un ensemble de poids d'amplitude fixe $\omega_{c1}, \omega_{c2}, \dots, \omega_{cN}$ qui produisent un signal non adapté. Ce formateur de voies conventionnel est identique à celui utilisé pour traiter les sorties du réseau de microphones avec des coefficients non adaptatifs fixes. Dans les applications typiques, les poids \mathbf{w}_c , sont choisis de manière à avoir un compromis entre la largeur du faisceau et le niveau des lobes secondaires. Ces poids sont souvent (mais pas nécessairement) définis par $\mathbf{w}_c = \mathbf{C}(\mathbf{C}^T \mathbf{C})^{-1} \mathbf{g}$. Un tel choix minimise sa norme $\mathbf{w}_c^T \mathbf{w}_c$, qui est également appelée le gain de bruit blanc.

Le deuxième bloc a pour objectif d'éliminer les lobes secondaires malgré que sa réponse est tout à fait le contraire. Ce bloc laisse passer tous les signaux mise à part le signal de la source désirée. Donc le second composant, \mathbf{w}_b qui est adapté, doit appartenir au sous-espace \mathcal{V} des filtres orthogonaux à la contrainte, où $\mathcal{V} = \{\mathbf{w}_b \in \mathbb{R}^{t,q} \mid \mathbf{C}^T \mathbf{w}_b = \mathbf{0}_{M \times 1}\}$.

Il se compose d'une matrice de blocage \mathbf{B} et un filtre adapté \mathbf{w}_b . Le but de \mathbf{B} est de bloquer le signal désiré $\mathbf{s}(k)$ du deuxième bloc, et le rôle du filtre FIR est de minimiser la puissance. Étant donné que $\mathbf{s}(k)$ est commun à chacune des sorties du réseau, le blocage est assuré si la somme des lignes de \mathbf{B} est égale à zéro. Si b_m^T est utilisé pour représenter la m ème ligne de \mathbf{B} , nous exigeons que b_m satisfait

$$b_m^T \mathbf{1} = 0, \quad \forall m \tag{2.53}$$

où $\mathbf{1}$ est un vecteur de taille $M \times 1$ qui ne contient que des uns.

Le signal désiré sera dans le premier bloc (le formateur de voies conventionnel), on note $y_c(k)$ la sortie du premier bloc. Dans le deuxième bloc on supprime le signal désiré avec la matrice de blocage \mathbf{B} , donc on aura que du bruit, on note $y_b(k)$ la sortie du premier bloc.

La sortie $y(k)$ sera

$$\mathbf{y}(k) = \mathbf{y}_c(k) - \mathbf{y}_b(k) \quad (2.54)$$

2.4.6 L'algorithme LCMV avec la structure GSC

Dans cette partie, on va développer la version GSC du formateur de voies à large bande, LCMV, expliquée par Griffiths et Jim dans leur article [14]. Cet structure est illustré dans la figure 2.9.

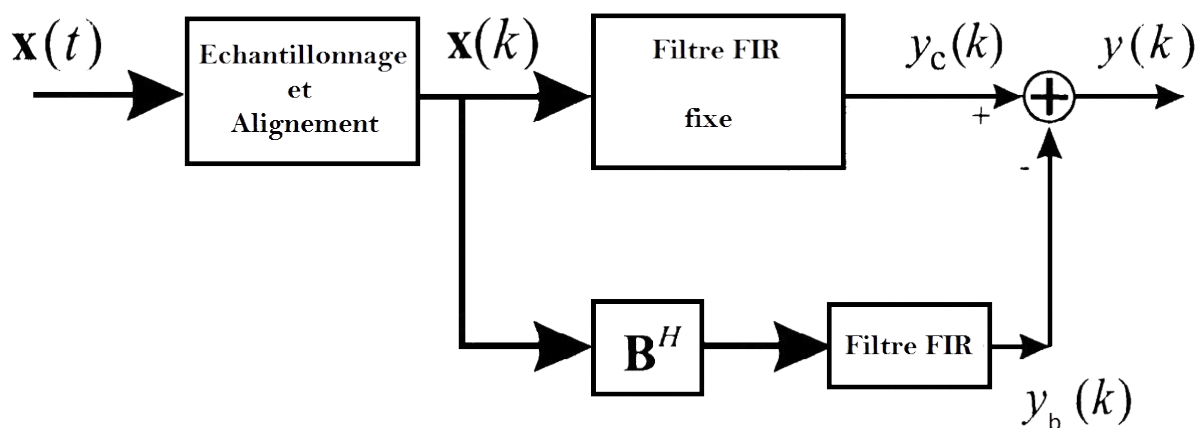


FIGURE 2.9 – L'implémentation de l'algorithme GSC.

Le premier bloc (le bloc supérieur de la figure 2.9) consiste en un formateur de voies avec un FIR de taille M , avec des poids w_i fixes, et une matrice de covariance $\mathbf{R}_{\mathbf{x}\mathbf{x}} = \mathbf{I}$,

$$\mathbf{w}_c^H = \mathbf{g}^H [\mathbf{C}^H \mathbf{C}]^{-1} \mathbf{C}^H \quad (2.55)$$

où \mathbf{g}^H et \mathbf{C} déjà définis en (2.50) et (2.51) respectivement. La sortie du premier bloc est $y_c(k)$.

La matrice de blocage \mathbf{B}^H dans le deuxième bloc est de taille $(N - N_c) \times N$ qui satisfait

$$\mathbf{C}_{N_c}^H \mathbf{B} = \mathbf{0}. \quad (2.56)$$

Où \mathbf{C}_{N_c} est la matrice de contrainte donnée dans (2.49), et $\mathbf{0}$ est une matrice ne contenant que des zéros de taille $N_c \times (N - N_c)$, où $(N - N_c)$ la taille de sous-espace orthogonal au sous-espace contraint.

On note \mathbf{x}_b la sortie de la matrice de blocage. A l'instant t_0 le vecteur d'entrée à la m ème colonne du filtre FIR est

$$\mathbf{x}_{bm} = \mathbf{x}_b(t_0 - mT_s) \quad (2.57)$$

On peut définir le vecteur \mathbf{x}_b de taille $(N - N_c) \times M$

$$\mathbf{x}_b = \begin{bmatrix} \mathbf{x}_{b0} \\ \mathbf{x}_{b1} \\ \vdots \\ \mathbf{x}_{b(M-1)} \end{bmatrix}, \quad (2.58)$$

Pour mettre en évidence la relation entre le vecteur d'entrée \mathbf{x} et le vecteur de sortie \mathbf{x}_b , on définit une matrice de taille $(N - N_c)M \times NM$ composée de la matrice \mathbf{B}_H

$$\mathbf{B}_{comp}^H = \begin{bmatrix} \mathbf{B}_H & 0 & \cdots & 0 \\ 0 & \mathbf{B}_H & & \vdots \\ \vdots & & \ddots & \\ 0 & 0 & \cdots & \mathbf{B}_H \end{bmatrix}, \quad (2.59)$$

Et

$$\mathbf{x}_b = \mathbf{B}_{comp}^H \mathbf{x} \quad (2.60)$$

Le filtre FIR dans le chemin inférieur est choisi pour minimiser la puissance de sortie qui est pour cette structure,

$$P_0 = [\mathbf{w}_c - \mathbf{B}_{comp} \mathbf{w}_b]^H \mathbf{R}_{xx} [\mathbf{w}_c - \mathbf{B}_{comp} \mathbf{w}_b]. \quad (2.61)$$

En prenant le gradient en respectant \mathbf{w}_b et en mettant le résultat égale à zero,

$$[\mathbf{w}_c - \mathbf{B}_{comp} \mathbf{w}_b]^H \mathbf{R}_{xx} \mathbf{B}_{comp} = 0, \quad (2.62)$$

on obtient le vecteur de taille $1 \times (N - N_c)M$

$$\mathbf{w}_b^H = \mathbf{w}_c^H \mathbf{R}_{xx} \mathbf{B}_{comp} \left[\mathbf{B}_{comp}^H \mathbf{R}_{xx} \mathbf{B}_{comp} \right]^{-1}.$$

La structure GSC est plus intéressante quand on l'implémente dans la version adaptative.

2.4.7 Le filtre LCMV avec le modèle réverbérant

La plupart des chambres sont réverbérantes, ce qui signifie que chaque microphone reçoit souvent un grand nombre d'échos en raison des reflets du front d'onde à partir d'objets et de limites de pièce comme les murs, le plafond et le sol. Dans ce modèle, les signaux reçus sont exprimés par

$$\begin{aligned} y_i(k) &= g_i * s(k) + n_i(k) \\ &= x_i(k) + n_i(k) \end{aligned} \quad (2.63)$$

Où g_i est la réponse impulsionnelle de la source inconnue $s(k)$ au i ème microphone. Nous supposons que $s(k)$ est de moyenne nulle, large bande et non corrélé avec le bruit additif $n_i(k)$. Dans une forme vecteur/matrice, le modèle de signal (2.63) peut être réécrit comme

$$y_i(k) = \mathbf{g}_i^T \mathbf{s}(k) + n_i(k), \quad i = 0, \dots, N-1 \quad (2.64)$$

Où

$$\begin{aligned} \mathbf{g}_i &= \begin{bmatrix} g_{i,0} & g_{i,1} & \dots & g_{i,L_g-1} \end{bmatrix}^T, \\ \mathbf{s}(k) &= \begin{bmatrix} s(k) & s(k-1) & \dots & s(k-L_g+1) \end{bmatrix}^T, \end{aligned} \quad (2.65)$$

Et L_g est la longueur des réponses impulsionnelles acoustiques des N canaux $g_i, i = 1, 2, \dots, N$.

Nous supposons que les réponses impulsionnelles de la source souhaitée aux microphones sont connues (ou peuvent être estimées) et sont stationnaires. Les signaux des microphones peuvent être réécrits sous la forme

$$\mathbf{y}_i(k) = \mathbf{G}_i \mathbf{s}_L(k) + \mathbf{n}_i(k), \quad i = 1, 2, \dots, N. \quad (2.66)$$

Où

$$\begin{aligned} \mathbf{y}_i(k) &= \begin{bmatrix} y_i(k) & y_i(k-1) & \dots & y_i(k-L_h+1) \end{bmatrix}^T, \\ \mathbf{n}_i(k) &= \begin{bmatrix} n_i(k) & n_i(k-1) & \dots & n_i(k-L_h+1) \end{bmatrix}^T, \\ \mathbf{s}_L(k) &= \begin{bmatrix} s(k) & s(k-1) & \dots & s(k-L+1) \end{bmatrix}^T, \end{aligned} \quad (2.67)$$

Et \mathbf{G}_i est de taille $L_h \times L$

$$\mathbf{G}_i = \begin{bmatrix} g_{i,0} & \dots & g_{i,L_g-1} & 0 & 0 & \dots & 0 \\ 0 & g_{i,0} & \dots & g_{i,L_g-1} & 0 & \dots & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & \dots & 0 & g_{i,0} & \dots & g_{i,L_g-1} \end{bmatrix} \quad (2.68)$$

avec $L = L_h + L_g - 1$,

Si on concatène les N vecteurs d'observation, on obtient :

$$\begin{aligned} \mathbf{y}(k) &= \left[\mathbf{y}_1^T(k) \quad \mathbf{y}_2^T(k-1) \quad \cdots \quad \mathbf{y}_N^T(k-L_h+1) \right]^T \\ &= \mathbf{G} \mathbf{s}_L(k) + \mathbf{n}(k) \end{aligned} \quad (2.69)$$

Où

$$\mathbf{G} = \begin{bmatrix} \mathbf{G}_1 \\ \mathbf{G}_2 \\ \vdots \\ \mathbf{G}_N \end{bmatrix}_{NL_h \times L} \quad (2.70)$$

$$\mathbf{n}(k) = \left[\mathbf{n}_1^T(k) \quad \mathbf{n}_2^T(k) \quad \cdots \quad \mathbf{n}_N^T(k) \right]. \quad (2.71)$$

La matrice de covariance de taille $NL_h \times NL_h$ correspondant à $\mathbf{y}(k)$ est

$$\mathbf{R}_{yy} = E \left[\mathbf{y}(k) \mathbf{y}^T(k) \right] = \mathbf{G} \mathbf{R}_{ss} \mathbf{G}^T + \mathbf{R}_{nn}, \quad (2.72)$$

avec \mathbf{R}_{ss} et \mathbf{R}_{nn} sont la matrice de covariance de source et la matrice de covariance de bruit respectivement.

Avec tous ces informations le formateur de voies LCMV est obtenu en résolvant le problème d'optimisation

$$\min \mathbf{w}^T \mathbf{R}_{yy} \mathbf{w} \quad \text{sous la contrainte} \quad \mathbf{G}^T \mathbf{w} = \mathbf{u} \quad (2.73)$$

où

$$\mathbf{u} = \left[1 \quad 0 \quad \cdots \quad 0 \right]^T$$

est un vecteur de longueur dont le premier composant est égal à 1 tandis que tous les autres sont des zéros. Les contraintes sont nécessaires pour la déréverbération du signal d'intérêt tandis que la minimisation est requise pour réduire le bruit.

Donc la solution optimale est

$$\mathbf{w} = \mathbf{R}_{yy}^{-1} \mathbf{G} (\mathbf{G}^T \mathbf{R}_{yy} \mathbf{G})^{-1} \mathbf{u} \quad (2.74)$$

Le filtre LCMV a le potentiel de réduire considérablement l'effet du bruit de fond avec un grand nombre de microphones. Le filtre LCMV avec le modèle réverbérant est très intéressant d'un point de vue théorique, car il permet, en général, une dé-réverbération parfaite (le signal désiré reste intacte) avec une grande quantité de réduction du bruit quand la valeur de L_h augmente de son minimum requis. Cependant, dans ce contexte, le filtre LCMV peut ne pas être très pratique puisque les réponses impulsionnelles

acoustiques de la source inconnue aux N microphones sont difficiles à estimer dans les applications du monde réel.

2.4.8 Le filtre LCMV avec le modèle spatio-temporel

Il semble que, afin d'éviter l'annulation du signal, nous devons assurer que nous dé-réverbérons parfaitement le signal d'intérêt. Cela nécessite la connaissance d'une quantité énorme d'informations, c'est-à-dire les N réponses impulsionnelles du signal d'intérêt pour les microphones, ce qui n'est pas très pratique à acquérir dans la plupart des applications. Il est alors juste de demander s'il est possible d'effectuer une réduction du bruit à l'un des signaux du microphone, $y_n(k)$, sans essayer de récupérer la source souhaitée $s(k)$? Le filtre LCMV développé avec le modèle spatio-temporel tente de le faire.

De l'équation 2.63 on a

$$x_i(k) = g_i * s(k) \quad (2.75)$$

On utilise la transformée en Z , on trouve,

$$X_i(z) = G_i(z)S(z) \quad (2.76)$$

D'après cette equation on peut voir que les $x_i(k)$ de $i = 2, \dots, N$ sont en relation avec $x_1(k)$

$$X_i(z) = \frac{G_i(z)}{G_1(z)}X_1(z) = H_i(z)X_1(z) \quad (2.77)$$

en revenant au domaine temporel en utilisant un filtre FIR suffisamment large

$$\mathbf{x}_i(k) = \mathbf{H}_i \mathbf{x}_1(k) \quad (2.78)$$

avec

$$\mathbf{x}_i(k) = \begin{bmatrix} x_i(k) & x_i(k-1) & \dots & x_i(k-L_h+1) \end{bmatrix}^T \quad i = 1, \dots, N$$

et \mathbf{H}_i est une matrice de taille $L_h \times L_h$, plus tard on va voir comment la déterminer.

Considérons maintenant le filtre de longueur $N \times L_h$ et la puissance de sortie $\mathbf{W}^T \mathbf{R}_{yy} \mathbf{W}$. Nous avons :

$$\mathbf{W}^T \mathbf{R}_{yy} \mathbf{W} = \mathbf{W}^T \mathbf{R}_{xx} \mathbf{W} + \mathbf{W}^T \mathbf{R}_{nn} \mathbf{W}$$

Où $\mathbf{R}_{xx} = E \left[\mathbf{x}(k) \mathbf{x}(k)^T \right]$ est la matrice de corrélation du signal

$$\mathbf{x}(k) = \begin{bmatrix} \mathbf{x}_1(k)^T & \mathbf{x}_2(k)^T & \dots & \mathbf{x}_N(k)^T \end{bmatrix}^T$$

On trouve

$$\mathbf{W}^T \mathbf{R}_{yy} \mathbf{W} = \mathbf{W}^T \mathbf{H} \mathbf{R}_{x_1 x_1} \mathbf{H}^T \mathbf{W} + \mathbf{W}^T \mathbf{R}_{nn} \mathbf{W} \quad (2.79)$$

Où

$$\mathbf{H} = \begin{bmatrix} \mathbf{I}_{L_h \times L_h} \\ \mathbf{H}_2 \\ \vdots \\ \mathbf{H}_N \end{bmatrix}$$

est une matrice de taille $NL_h \times L_h$

On a la contrainte $\mathbf{H}^T \mathbf{W} = \mathbf{u}'$, l'équation (2.79) devient

$$\mathbf{W}^T \mathbf{R}_{yy} \mathbf{W} = \sigma_{x_1}^2 + \mathbf{W}^T \mathbf{R}_{nn} \mathbf{W} \quad (2.80)$$

Où

$$\mathbf{u}' = \begin{bmatrix} 1 & 0 & \cdots & 0 \end{bmatrix}^T,$$

L'expression (2.80) montre clairement qu'il est possible de récupérer $\mathbf{x}_1(k)$ sans distorsion tout en réduisant le bruit.

En résolvant le problème d'optimisation

$$\min \mathbf{W}^T \mathbf{R}_{yy} \mathbf{W} \quad \text{avec} \quad \mathbf{H}^T \mathbf{W} = \mathbf{u}',$$

Le formateur de voies optimal est

$$\mathbf{W} = \mathbf{R}_{yy}^{-1} \mathbf{H} (\mathbf{H}^T \mathbf{R}_{yy} \mathbf{H})^{-1} \mathbf{u}'$$

Maintenant on doit déterminer la matrice \mathbf{H} , une estimation optimale peut être obtenue en minimisant la fonction de cout suivante

$$J(\mathbf{W}_n) = E\{[\mathbf{x}_n(k) - \mathbf{H}_n(k)\mathbf{x}_1(k)]^T [\mathbf{x}_n(k) - \mathbf{H}_n(k)\mathbf{x}_1(k)]\}$$

On trouve la solution optimale

$$\mathbf{H}_n = \mathbf{R}_{\mathbf{x}_n \mathbf{x}_1} \mathbf{R}_{\mathbf{x}_1 \mathbf{x}_1}^{-1}$$

Où $\mathbf{R}_{\mathbf{x}_n \mathbf{x}_1} = E[\mathbf{x}_n(k)\mathbf{x}_1(k)^T]$ est la matrice de la corrélation croisées entre les signaux. Cependant, le signal de la source, $\mathbf{x}_n(k)$ $n = 1, 2, \dots, N$, est non observable, donc on doit trouver une autre manière pour estimer la matrice \mathbf{H} . En utilisant, $\mathbf{x}_n(k) = \mathbf{y}_n(k) - \mathbf{n}_n(k)$, on peut vérifier que,

$$\mathbf{R}_{\mathbf{x}_n \mathbf{x}_1} = \mathbf{R}_{\mathbf{y}_n \mathbf{y}_1} - \mathbf{R}_{\mathbf{n}_n \mathbf{n}_1}, \quad n = 1, 2, \dots, N$$

Où $\mathbf{R}_{\mathbf{y}_n \mathbf{y}_1} = E[\mathbf{y}_n(k)\mathbf{y}_1^T(k)]$ et $\mathbf{R}_{\mathbf{n}_n \mathbf{n}_1} = E[\mathbf{n}_n(k)\mathbf{n}_1^T(k)]$, donc

$$\mathbf{H}_n = (\mathbf{R}_{\mathbf{y}_n \mathbf{y}_1} - \mathbf{R}_{\mathbf{n}_n \mathbf{n}_1})(\mathbf{R}_{\mathbf{y}_1 \mathbf{y}_1} - \mathbf{R}_{\mathbf{n}_1 \mathbf{n}_1})^{-1}$$

Maintenant le filtre optimal dépend seulement des statistiques du second ordre du signal et du bruit. les statistiques du bruit doivent être estimées dans la période du silence (où $\mathbf{s}(k) = 0$) si nous supposons que le signal de bruit est stationnaire, et donc, on peut réutiliser ces statistiques quand le signal sera actif.

2.4.9 La méthode LS (Least Squares)

La méthode LS appartient à la classe de techniques 'fixed beamforming', qui prend en compte la géométrie du réseau mais ne suppose aucune information de la source ni des signaux de bruit [15].

Dans la méthode LS, l'objectif est d'optimiser les coefficients du formateur de voies de sorte que la réponse directionnelle résultante puisse mieux se rapprocher d'une réponse directionnelle donnée.

Compte tenu d'une réponse désirée $D(f, \theta)$, le critère d'approximation LS pour la conception d'un formateur de voies large bande général peut être formulée de manière simple comme suit

$$J_{LS} = \int_{B_s} \int_{\Theta} F(f, \theta) |\mathbf{w}^H \mathbf{a}(f, \theta) - D(f, \theta)|^2 df d\theta \quad (2.81)$$

Où \mathbf{w} est le vecteur de pondération défini dans (2.34) de taille $MN \times 1$ et

$$\mathbf{a} = \begin{bmatrix} \mathbf{a}_0(f, \theta) \\ \vdots \\ \mathbf{a}_m(f, \theta) \\ \vdots \\ \mathbf{a}_{M-1}(f, \theta) \end{bmatrix}, \quad (2.82)$$

$\mathbf{a}_m(f, \theta)$ est le vecteur de direction défini dans (2.32), de taille $N \times 1$, B_s et Θ représentent respectivement la bande de fréquence de la parole et la plage d'angle de direction, et $F(f, \theta)$ est une fonction de pondération positive.

Considérons le cas de conception spécifique avec $F(f, \theta) = 1$ et $D(f, \theta) = 1$ dans la bande passante, et $F(f, \theta) = \alpha$ et $D(f, \theta) = 0$ dans la bande interdite. Pour calculer l'intégration dans l'équation ci-dessus, nous pouvons échantillonner uniformément les plages de fréquence et d'angle. Soit (f_n, θ_n) désignent un point de prélèvement arbitraire. Ensuite, la formulation LS passe à

$$J_{LS} = \sum_{f_n \in B_s} \sum_{\theta_k \in \Theta_m} |\mathbf{w}^H \mathbf{v}(f_n, \theta_k) - 1|^2 + \alpha \sum_{f_n \in B_s} \sum_{\theta_k \in \Theta_s} |\mathbf{w}^H \mathbf{v}(f_n, \theta_k)|^2 \quad (2.83)$$

Où Θ_s et Θ_m représentent respectivement la région des lobes secondaires et la région du lobe principal.

L'équation (2.83) peut être réécrite dans une forme quadratique

$$J_{LS} = \mathbf{w}^H \mathbf{Q}_{LS} \mathbf{w} - \mathbf{w}^H \mathbf{a} - \mathbf{a}^H \mathbf{w} + d_{LS} \quad (2.84)$$

avec

$$\mathbf{Q}_{LS} = \sum_{f_n \in B_s} \sum_{\theta_k \in \Theta_m} \mathbf{A}(f_n, \theta_k) + \alpha \sum_{f_n \in B_s} \sum_{\theta_k \in \Theta_s} \mathbf{A}(f_n, \theta_k).$$

$$\mathbf{v} = \sum_{f_n \in B_s} \sum_{\theta_k \in \Theta_m} \mathbf{a}(f_n, \theta_k).$$

$$d_{LS} = \sum_{f_n \in B_s} \sum_{\theta_k \in \Theta_m} 1.$$

Où $\mathbf{A}(f_n, \theta_k) = \mathbf{a}^H(f_n, \theta_k) \mathbf{a}(f_n, \theta_k)$

La minimisation de la fonction de coût dans (2.84) va nous donner le formateur de voies LS

$$\mathbf{w}_{LS} = \mathbf{Q}_{LS}^{-1} \mathbf{a} \quad (2.85)$$

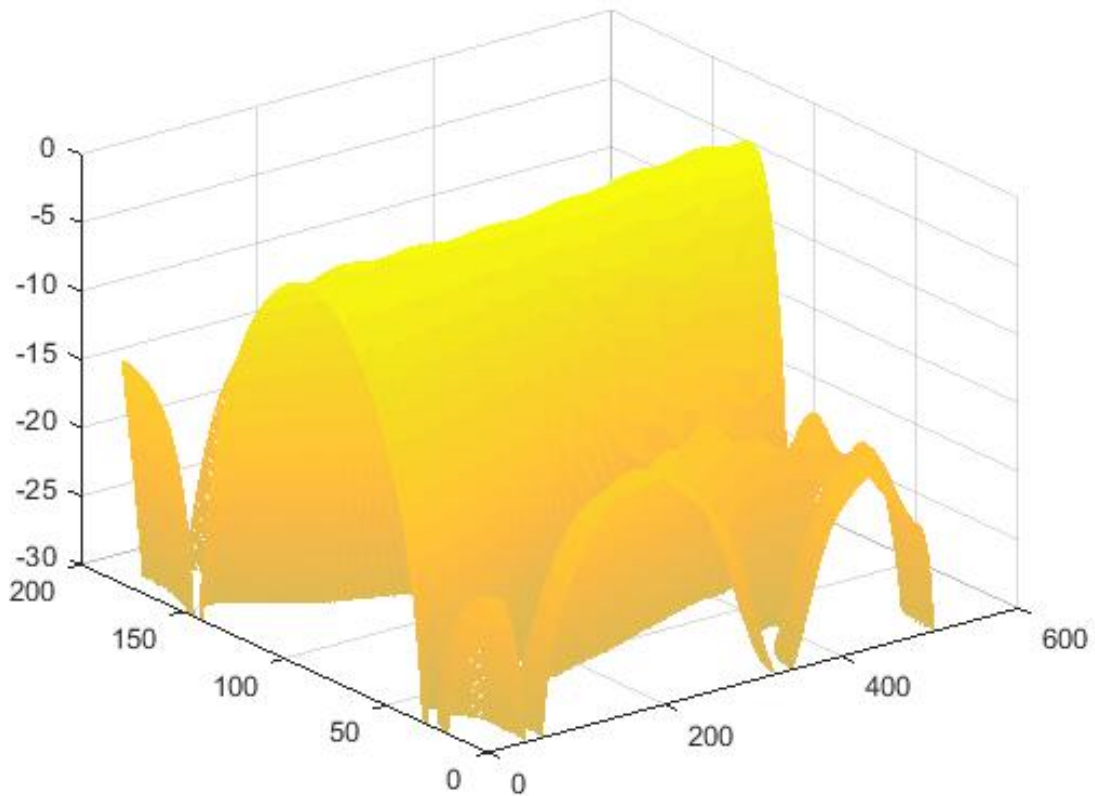


FIGURE 2.10 – Le diagramme de directivité du filtre LS.

La figure 2.10 représente la réponse fréquentielle de formateur de voies LS, on voit que c'est une bonne réponse avec des lobes secondaires beaucoup moins important que le lobe principale, le premier lobe secondaire est à -16 dB, qui est un rapport de 40. Mais elle reste toujours dépendante de fréquence donc ce n'est pas tout à fait une bonne méthode à large bande.

2.5 Les formateurs de voies adaptatifs

Dans cette section, nous développons des techniques pour la mise en œuvre des algorithmes développés précédemment en utilisant les données entrantes. Nous avons supposé que nous connaissions les directions du signal et la matrice de covariance \mathbf{R}_{xx} . Dans les applications réelles, nous devons estimer ces quantités des données entrantes. Pour la direction d'arrivée, il existe plusieurs méthodes comment l'estimer, pour la matrice de covariance, elle doit être estimée des données entrantes. Le formateur de voies adaptatif s'adapte aux données entrantes dès qu'un échantillon arrive. Dans ce chapitre, nous développons la théorie et la pratique des formateurs de voies adaptatifs.

Les filtres adaptatifs ont le potentiel de s'adapter au changement de l'environnement, nous allons démontrer les performances de ces filtres dans les résultats expérimentaux.

2.5.1 Estimation de la matrice de covariance

On a les échantillons disponibles des données entrantes, $\mathbf{x}(1), \mathbf{x}(0), \dots, \mathbf{x}(K)$, où $\mathbf{x}(k)$ de taille $NM \times 1$ est le k ème échantillon.

donc

$$\hat{\mathbf{R}}_{xx} = \frac{1}{K} \sum_{k=1}^K \mathbf{x}(k)\mathbf{x}(k)^H \quad (2.86)$$

Où K est la taille du signal échantillonné.

2.5.2 L'implémentation direct

Soit le formateur de voies de Frost expliqué dans 2.4.3

$$\mathbf{w}_{mvd}^H = \mathbf{g}^H \left[\mathbf{C}^H \hat{\mathbf{R}}_x^{-1} \mathbf{C} \right] \mathbf{C}^H \hat{\mathbf{R}}_x^{-1} \quad (2.87)$$

Dans l'implémentation direct la matrice de covariance est estimée des données échantillonnées, d'après (2.86),

$$\hat{\mathbf{R}}_{xx} = \frac{1}{K} \sum_{k=1}^K \mathbf{x}(k)\mathbf{x}(k)^H, \quad (2.88)$$

Où $\mathbf{x}(k)$ est un vecteur d'échantillons de taille $NM \times 1$.

L'implémentation direct exige une complexité énorme de calcul, on ne peut pas l'implémenter dans une application qui fonctionne en temps réel. Aussi, elle ne s'adapte pas au changement de l'environnement.

2.5.3 L'implémentation LMS

Dans cette partie, on étudie la version adaptative de l'algorithme de Frost, ou, (Constrained Least Mean-Squares), Frost a utilisé la méthode adaptatif LMS pour son algorithme.

Constrained LMS est un algorithme adaptatif, simple, pour ajuster un réseau de microphones en temps réel pour répondre à un signal désiré provenant d'une direction donnée, et en filtrant les signaux provenant des autres directions. L'algorithme adapte de façon itérative les poids du formateur de voies pour minimiser la puissance du bruit en sortie du réseau en gardant une réponse fréquentielle dans la direction d'observation.

Dans le processus adaptatif, l'algorithme apprend progressivement des statistiques concernant le bruit provenant d'autres directions que la direction d'observation. Les bruits arrivant de la direction d'observation peuvent être filtrés par un choix approprié de la caractéristique de réponse en fréquence dans cette direction.

Un avantage majeur de l'algorithme Constrained LMS est qu'il dispose d'une fonction d'auto-correction qui lui permet de fonctionner pour des périodes de temps arbitrairement longues dans une implémentation informatique numérique sans se détourner de ses contraintes en raison des erreurs cumulatives.

La substitution directe de la matrice de corrélation à l'équation du poids optimal (2.87) nécessite un nombre de multiplications à chaque itération proportionnelle au cube du nombre de poids. Ceci principalement causé par l'inversion requise de la matrice de corrélation \mathbf{R}_{xx} .

Frost, dans son article a bien expliqué les détails, ici, on va seulement donner les étapes nécessaires, Soit la méthode adaptative 'steepest gradient-descent'

$$\mathbf{w}(k+1) = \mathbf{w}(k) - \mu \frac{\partial \mathcal{L}(\mathbf{w}(k), \lambda)}{\partial \mathbf{w}(k)} \quad (2.89)$$

Où $\frac{\partial \mathcal{L}(\mathbf{w}(k), \lambda)}{\partial \mathbf{w}(k)}$ est défini dans (2.45)

donc

$$\mathbf{w}(k+1) = \mathbf{w}(k) - \mu(\mathbf{w}^H(k)\mathbf{R}_{xx} + \lambda^H \mathbf{C}^H). \quad (2.90)$$

Et en remplaçant (2.90) dans la contrainte

$$\mathbf{g} = \mathbf{C}^H \mathbf{w}(k+1) = \mathbf{C}^H (\mathbf{w}(k) - \mu(\mathbf{w}^H(k)\mathbf{R}_{xx} + \lambda^H \mathbf{C}^H)) \quad (2.91)$$

Avec (2.90) et (2.91), on résolve par rapport à λ^H on trouve,

$$\mathbf{w}(k+1) = \mathbf{w}(k) - \mu \left[\mathbf{I} - \mathbf{C}(\mathbf{C}^T \mathbf{C})^{-1} \mathbf{C}^T \right] \mathbf{R}_{xx} \mathbf{w}(k) + \mathbf{C}(\mathbf{C}^T \mathbf{C})^{-1} \left[\mathbf{g} - \mathbf{C}^T \mathbf{w}(k) \right] \quad (2.92)$$

On définit le vecteur de taille NM

$$\mathbf{F} = \mathbf{C}(\mathbf{C}^T \mathbf{C})^{-1} \mathbf{g} \quad (2.93)$$

et la matrice de taille $NM \times NM$

$$\mathbf{P} = \mathbf{I} - \mathbf{C}(\mathbf{C}^T \mathbf{C})^{-1} \mathbf{C}^T \quad (2.94)$$

Où \mathbf{I} est matrice identité.

L'algorithme peut être écrit comme

$$\mathbf{w}(k+1) = \mathbf{P} [\mathbf{w}(k) - \mu \mathbf{R}_{\mathbf{xx}} \mathbf{w}(k)] + \mathbf{F}. \quad (2.95)$$

Où μ est le pas d'adaptation qui doit être choisi pour garder la convergence :

$$0 < \mu < \frac{2}{\lambda_{max}}$$

Et λ_{max} est la plus grande valeur propre de $\mathbf{R}_{\mathbf{xx}}$

L'équation (2.95) exige la connaissance de $\mathbf{R}_{\mathbf{xx}}$, qui, n'est pas disponible a priori. Une approximation de cette matrice dans la k ème itération est $\mathbf{x}(k)\mathbf{x}(k)^T$. La substitution de cette estimation dans (2.95) nous donne l'algorithme Constrained LMS,

$$\mathbf{w}(0) = \mathbf{F}$$

$$\mathbf{w}(k+1) = \mathbf{P} [\mathbf{w}(k) - \mu \mathbf{y}(k) \mathbf{x}(k)] + \mathbf{F} \quad (2.96)$$

L'algorithme Constrained LMS satisfait la contrainte $\mathbf{C}^T \mathbf{w}(k+1) = \mathbf{g}$ à chaque itération. A chaque itération, l'algorithme nécessite uniquement $\mathbf{x}(k)$ et la sortie de réseau $y(k)$. Aucune connaissance a priori de la matrice de corrélation d'entrée n'est nécessaire. \mathbf{F} est un vecteur constant qui peut être calculer une seule fois. L'une des deux opérations les plus complexes requises par (2.96) est la multiplication de chacun des NM composants du vecteur $\mathbf{x}(k)$ par l'élément scalaire $\mu \mathbf{y}(k)$, L'autre opération significative est indiquée par la matrice \mathbf{P} .

2.5.4 L'implémentation GSC

Soit la structure GSC expliqué dans (2.4.5), l'implémentation adaptative de l'algorithme de Frost avec la structure GSC est illustré dans la Figure 2.9. On note $\mathbf{z}(k)$ la sortie de la matrice de blocage \mathbf{B}_{comp}

$$\mathbf{z}(k) = \mathbf{B}_{comp} \mathbf{x}_b$$

On a aussi

$$\mathbf{y}_c(k) = \mathbf{w}_q^H \mathbf{x}(k)$$

Et d'après les équations de l'algorithme adaptatif LMS

$$\mathbf{e}(k) = \mathbf{y}_c(k) - \mathbf{w}_a^H(k-1)\mathbf{z}(k)$$

$$\mathbf{w}_a(k) = \mathbf{w}_a(k-1) + \Delta(k)\mathbf{z}(k)\mathbf{e}(k)$$

Et

$$\mathbf{w} = \mathbf{w}_q - \mathbf{B}_{comp}\mathbf{w}_a$$

avec

$$\Delta(k) = \frac{\mu}{\sigma_x^2}$$

Où μ est le pas, et σ_x^2 est la variance du signal d'entrée.

2.5.5 L'implémentation de l'algorithme TF-GSC

Sharon Gannot a présenté un algorithme basé sur la structure GSC [16], et qui peut travailler et donner une bonne performance de filtrage de bruit, avec le signal de la parole et aussi même dans un environnement réverbérant. Cet algorithme prend en considération la fonction de transfert (TF) entre la source du signal et le réseau de microphone. La fonction de transfert entre la source de signal et chaque microphone du réseau est estimée en exploitant la non-stationnarité du signal de la parole. Ce qui rend cet algorithme plus intéressant est qu'on peut même implémenter la version adaptatif de l'algorithme.

Dans l'algorithme de Frost, et l'algorithme GSC développé par Griffiths, on suppose que le signal à chaque microphone est une version décalée du signal de la source désiré. Dans cet algorithme on considère le cas où la fonction de transfert TF (Transfert Function) entre la source et le réseau de microphones est quelconque.

2.5.5.1 Le modèle de signal

On considère un réseau de microphones dans un environnement réverbérant et bruité, on utilise les notations suivantes :

$z_m(t)$: le signal dans le m ème microphone.

$s(t)$: le signal désiré.

$n_m(t)$: le bruit dans le m ème microphone.

$a_m(t)$: la fonction de transfert (TF) entre la source et le m ème microphone.

Donc on a

$$z_m(t) = a_m(t) * s(t) + n_m(t), \quad m = 1, \dots, M.$$

Où M est le nombre de microphones du réseau, et $*$ est la convolution. Supposant que la fenêtre d'analyse est choisi de telle sorte que le signal soit stationnaire, on multiplie les deux cotés par une fenêtre rectangulaire, et on applique la transformée de fourrier à courte terme (STFT, Short-Term-Fourier-Transform),

on obtient,

$$Z_m(t, e^{jw}) = A_m(e^{jw})S(t, e^{jw}) + N_m(t, e^{jw}) \quad (2.97)$$

$$m = 1, \dots, M$$

Si on formule l'équation (1) sous forme de vecteur on obtient,

$$\mathbf{Z}(t, e^{jw}) = \mathbf{A}(e^{jw})S(t, e^{jw}) + \mathbf{N}(t, e^{jw}) \quad (2.98)$$

Où

$$\begin{aligned} \mathbf{Z}(t, e^{jw}) &= \begin{bmatrix} Z_1(t, e^{jw}) & Z_2(t, e^{jw}) & \dots & Z_M(t, e^{jw}) \end{bmatrix} \\ \mathbf{A}(e^{jw}) &= \begin{bmatrix} A_1(e^{jw}) & A_2(e^{jw}) & \dots & A_M(e^{jw}) \end{bmatrix} \\ \mathbf{N}(t, e^{jw}) &= \begin{bmatrix} N_1(t, e^{jw}) & N_2(t, e^{jw}) & \dots & N_M(t, e^{jw}) \end{bmatrix} \end{aligned}$$

Dans l'algorithme de Frost on n'a pas pris en compte la fonction de transfert entre la source et les microphones. On va commencer par donner l'algorithme de frost dans le domaine fréquentiel, et après, on fait l'extension à l'algorithme (TF-GSC) qui prend en considération la fonction de transfert.

2.5.5.2 L'algorithme de frost dans le domaine fréquentiel

Soit la solution optimal de l'algorithme de Frost

$$\mathbf{W}^H(t, e^{jw}) = \begin{bmatrix} W_1^*(t, e^{jw}) & W_2^*(t, e^{jw}) & \dots & W_M^*(t, e^{jw}) \end{bmatrix}$$

Où * est le conjugué. La sortie est

$$\begin{aligned} Y(t, e^{jw}) &= \mathbf{W}^H(t, e^{jw})\mathbf{Z}(t, e^{jw}) \\ &= \mathbf{W}^H(t, e^{jw})\mathbf{A}(e^{jw})S(t, e^{jw}) + \mathbf{W}^H(t, e^{jw})\mathbf{N}(t, e^{jw}) \\ &= Y_s(t, e^{jw}) + Y_n(t, e^{jw}) \end{aligned} \quad (2.99)$$

Où $Y_s(t, e^{jw})$ est la partie du signal désiré, et $Y_n(t, e^{jw})$ est la partie du bruit. La puissance de sortie de ce formateur de voies est

$$\begin{aligned} E\{Y(t, e^{jw})Y^*(t, e^{jw})\} &= E\{\mathbf{W}^H(t, e^{jw})\mathbf{Z}(t, e^{jw})\mathbf{Z}^H(t, e^{jw})\mathbf{W}(t, e^{jw})\} \\ &= \mathbf{W}^H(t, e^{jw})\Phi_{\mathbf{ZZ}}(t, e^{jw})\mathbf{W}(t, e^{jw}) \end{aligned} \quad (2.100)$$

Où $\Phi_{\mathbf{ZZ}}(t, e^{jw}) = E\{\mathbf{Z}(t, e^{jw})\mathbf{Z}^H(t, e^{jw})\}$. Maintenant, on veut minimiser la puissance de sortie en gardant la contrainte suivante sur la sortie,

$$\begin{aligned} Y_s(t, e^{jw}) &= \mathbf{W}^H(t, e^{jw})\mathbf{A}(e^{jw})S(t, e^{jw}) \\ &= g^*(t, e^{jw})S(t, e^{jw}) \end{aligned} \quad (2.101)$$

Où $g^*(t, e^{jw})$ est un filtre spécifié (il peut être un simple décalage). Donc, on résolve le problème d'optimisation suivant :

$$\min \mathbf{W}^H(t, e^{jw}) \Phi_{ZZ}(t, e^{jw}) \mathbf{W}(t, e^{jw}) \quad \text{avec} \quad \mathbf{W}^H(t, e^{jw}) \mathbf{A}(e^{jw}) = g^*(t, e^{jw})$$

En utilisant le multiplicateur de Lagrange, le filtre optimal est

$$\mathbf{W}(t, e^{jw})_{opt} = \left[\mathbf{A}^H(e^{jw}) \Phi_{ZZ}^{-1}(t, e^{jw}) \mathbf{A}(e^{jw}) \right]^{-1} \Phi_{ZZ}^{-1}(t, e^{jw}) \mathbf{A}(e^{jw}) g(t, e^{jw})$$

Cet algorithme est difficile à implémenter, et il ne peut pas s'adapter au changement de l'environnement, une version adaptative doit être applicable.

En appliquant la méthode adaptative LMS expliqué précédemment,

$$\mathbf{W}(t+1, e^{jw}) = P(e^{jw}) \left[\mathbf{W}(t, e^{jw}) - \mu \mathbf{Z}(t, e^{jw}) Y^*(t, e^{jw}) \right] + \mathbf{F}(e^{jw})$$

Où

$$P(e^{jw}) = I - \frac{\mathbf{A}(e^{jw}) \mathbf{A}^H(e^{jw})}{\|\mathbf{A}(e^{jw})\|^2}$$

et

$$\mathbf{F}(e^{jw}) = \frac{\mathbf{A}(e^{jw})}{\|\mathbf{A}(e^{jw})\|^2} g(e^{jw})$$

2.5.5.3 Generalized Sidelobe Canceller

Griffiths et Jim ont développé la structure GSC, et ont implémenté l'algorithme de Frost avec cette structure. Ils ont pris le cas où la fonction de transfert (TF) est un décalage. Ils ont aussi implémenté la version adaptative de l'algorithme pour diminuer la complexité de calcul.

Nous allons développer l'algorithme GSC en prenant en considération la fonction de transfert entre la source et le réseau. Dans la Figure 2.11, on a la structure de l'algorithme TF-GSC, dans cette section nous allons développer la version adaptative LMS, de l'algorithme TF-GSC.

L'algorithme GSC, est divisé en trois composants :

1. Le formateur de voies fix (fixed beamformer(*FBF*)) \mathbf{W}_0^H , en haut de la figure 2.11.
2. La matrice de blocage $\mathcal{H}(e^{jw})$.
3. Le filtre adaptatif pour supprimer le bruit (noise canceler (*NC*)).

On va discuter chaque composant de l'algorithme, on a en sortie du formateur de voies

$$Y(t, e^{jw}) = Y_{FBF}(t, e^{jw}) - Y_{NC}(t, e^{jw}). \quad (2.102)$$

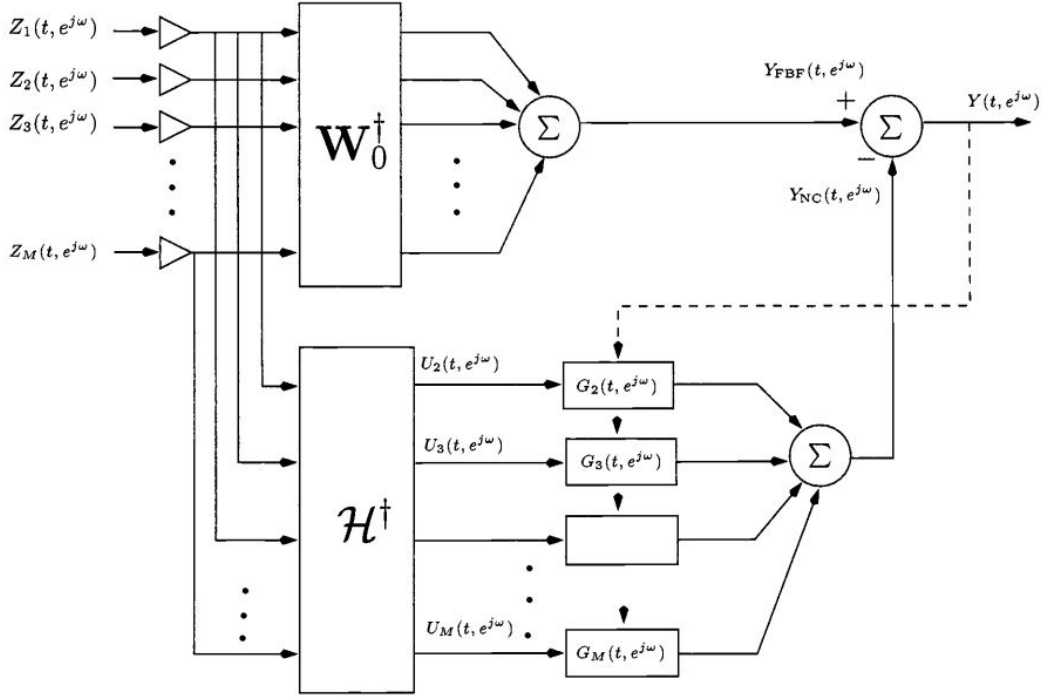


FIGURE 2.11 – La structure de l’algorithme TF-GSC.

1- Le formateur de voies fixe (FBF) : la structure GSC impose que le chemin supérieur passe tous le signal désiré provenant du réseau, et de l’algorithme de Frost on choisi la matrice $\Phi_{ZZ}(t, e^{j\omega}) = \mathbf{I}$, où \mathbf{I} est matrice identité. Donc

$$\mathbf{W}_0(t, e^{j\omega}) = \mathbf{F}(e^{j\omega}) = \frac{\mathbf{A}(e^{j\omega})}{\|\mathbf{A}(e^{j\omega})\|^2} g(e^{j\omega}). \quad (2.103)$$

Et

$$Y_{FBF} = \mathbf{W}_0^H(t, e^{j\omega}) \mathbf{Z}(t, e^{j\omega}).$$

$$Y_{FBF} = g^*(e^{j\omega}) S(t, e^{j\omega}) + g^*(e^{j\omega}) \frac{\mathbf{A}^H(e^{j\omega})}{\|\mathbf{A}(e^{j\omega})\|^2} \mathbf{N}(t, e^{j\omega}).$$

Le premier terme dans le côté droit est le terme du signal désiré, le seconde, est le terme du bruit.

Malheureusement, on n’a pas accès aux fonctions de transferts $A_m(e^{j\omega})$, $m = 1, \dots, M$, plus tard, on va voir comment les estimer.

Soit le rapport de fonction de transfert(TFs),

$$H_m(e^{j\omega}) = \frac{A_m(e^{j\omega})}{A_1(e^{j\omega})}$$

et

$$\mathbf{H}^T(e^{j\omega}) = \left[1 \quad \frac{A_2(e^{j\omega})}{A_1(e^{j\omega})} \quad \dots \quad \frac{A_M(e^{j\omega})}{A_1(e^{j\omega})} \right] \quad (2.104)$$

Dans (2.103), on remplace les nouvelles TFs, on obtient,

$$\mathbf{W}_0(t, e^{j\omega}) = \frac{\mathbf{H}(e^{j\omega})}{\|\mathbf{H}(e^{j\omega})\|^2} g(e^{j\omega}). \quad (2.105)$$

2- La matrice de blocage $\mathcal{H}(e^{jw})$: Cette matrice supprime le signal désiré et elle ne laisse que le bruit, cette matrice de taille $M \times (M - 1)$ doit satisfaire,

$$\mathbf{A}^H(e^{jw})\mathcal{H}(e^{jw}) = 0 \quad (2.106)$$

donc

$$\mathcal{H}(e^{jw}) = \begin{bmatrix} -\frac{A_2^*(e^{jw})}{A_1^*(e^{jw})} & -\frac{A_3^*(e^{jw})}{A_1^*(e^{jw})} & \cdots & -\frac{A_M^*(e^{jw})}{A_1^*(e^{jw})} \\ 1 & 0 & \cdots & 0 \\ 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & 1 \end{bmatrix} \quad (2.107)$$

3- Le filtre adaptatif pour supprimer le bruit (noise canceler) : De la Figure 2.11,

$$Y_{NC}(t, e^{jw}) = \mathbf{G}^H(t, e^{jw})\mathcal{H}^H(t, e^{jw})\mathbf{Z}(t, e^{jw}). \quad (2.108)$$

On note la sortie de la matrice de blocage, $\mathbf{U}(t, e^{jw})$, tel que,

$$\begin{aligned} \mathbf{U}(t, e^{jw}) &= \mathcal{H}^H(t, e^{jw})\mathbf{Z}(t, e^{jw}) \\ &= \mathcal{H}^H(t, e^{jw}) \left[\mathbf{A}(e^{jw})S(t, e^{jw}) + \mathbf{N}(t, e^{jw}) \right] \\ &= \mathcal{H}^H(t, e^{jw})\mathbf{N}(t, e^{jw}) \end{aligned} \quad (2.109)$$

On voit bien que la sortie de la matrice de blocage ne représente que le bruit. La matrice \mathbf{G} est utilisé pour ajuster la sortie Y_{NC} pour que la puissance de sortie soit minimale, et pour réduire le bruit contenu dans le signal Y_{BBF} .

Pour déterminer la matrice \mathbf{G} , on doit minimiser la puissance en sortie du formateur de voies. On minimise la puissance de sortie par rapport à $\mathbf{G}(t, e^{jw})$,

$$E \left[\|Y_{BBF}(t, e^{jw}) - \mathbf{G}^H(t, e^{jw})\mathbf{U}(t, e^{jw})\|^2 \right]$$

on trouve,

$$\mathbf{G}(t, e^{jw}) = \Phi_{\mathbf{UU}}^{-1}(t, e^{jw})\Phi_{\mathbf{UY}}(t, e^{jw}) \quad (2.110)$$

Où,

$$\begin{aligned} \Phi_{\mathbf{UY}}(t, e^{jw}) &= E \left[\mathbf{U}(t, e^{jw})Y_{BBF}^*(t, e^{jw}) \right] \\ \Phi_{\mathbf{UU}}(t, e^{jw}) &= E \left[\mathbf{U}(t, e^{jw})\mathbf{U}^H(t, e^{jw}) \right] \end{aligned}$$

Pour pouvoir capter le changement de l'environnement on traite le signal par segments. On utilise l'algorithme adaptatif LMS,

$$\mathbf{G}(t+1, e^{jw}) = \mathbf{G}(t, e^{jw}) + \mu\mathbf{U}(t, e^{jw})Y^*(t, e^{jw})$$

A ce stade, on a considéré que les rapports de fonctions de transferts (TFs) sont connus a priori, dans la partie suivante on va expliquer comment estimer ces rapports.

2.5.5.4 Estimation des rapports de fonctions de transferts (TFs)

De l'équation (2.109), on déduit,

$$Z_m(t, e^{jw}) = H_m(e^{jw})Z_1(e^{jw}) + U_m(t, e^{jw})$$

Considérons un intervalle d'analyse pendant lequel les TFs et le signal du bruit sont supposés stationnaires. Nous divisons cet intervalle d'analyse en trames de sorte que le signal souhaité peut être considéré comme stationnaire au cours de chaque trame. Considérons la k ème trame,

$$\Phi_{z_m z_1}^{(k)}(e^{jw}) = H_m(e^{jw})\Phi_{z_1 z_1}^{(k)}(e^{jw}) + \Phi_{u_m z_1}(e^{jw}) \quad (2.111)$$

$$k = 1, \dots, K.$$

Où $\Phi_{z_i z_j}^{(k)}(e^{jw})$ est la densité spectrale de puissance croisée (cross-PSD) entre z_i et z_j , et $\Phi_{u_m z_1}(e^{jw})$ est la densité spectrale de puissance croisée (cross-PSD) entre u_m et z_1

On a le signal $S(t, e^{jw})$ et le bruit $N(t, e^{jw})$ sont indépendants donc $\Phi_{u_m z_1}(e^{jw})$ est indépendant de la trame k .

Si le signal $Z_1(t, e^{jw})$, et le signal de bruit $U_m(t, e^{jw})$ sont dé-corrélés, donc, $H_m(e^{jw}) = \frac{\Phi_{z_m z_1}(e^{jw})}{\Phi_{z_1 z_1}(e^{jw})}$, mais en réalité, ils sont corrélés, si on réécrit (2.111) sous forme matricielle, O. Shalvi et E. Weinstein [17] a donné une solution en utilisant la méthode least squares,

$$\begin{bmatrix} \Phi_{z_m z_1}^{(1)}(e^{jw}) \\ \Phi_{z_m z_1}^{(2)}(e^{jw}) \\ \vdots \\ \Phi_{z_m z_1}^{(K)}(e^{jw}) \end{bmatrix} = \begin{bmatrix} \Phi_{z_1 z_1}^{(1)}(e^{jw}) & 1 \\ \Phi_{z_1 z_1}^{(2)}(e^{jw}) & 1 \\ \vdots & \vdots \\ \Phi_{z_1 z_1}^{(K)}(e^{jw}) & 1 \end{bmatrix} \begin{bmatrix} H_m(e^{jw}) \\ \Phi_{u_m z_1}(e^{jw}) \end{bmatrix} \quad (2.112)$$

La solution est,

$$H_m(e^{jw}) = \frac{\langle \Phi_{z_1 z_1}(e^{jw})\Phi_{z_m z_1}(e^{jw}) \rangle - \langle \Phi_{z_1 z_1}(e^{jw}) \rangle \langle \Phi_{z_m z_1}(e^{jw}) \rangle}{\langle \Phi_{z_1 z_1}^2(e^{jw}) \rangle - \langle \Phi_{z_1 z_1}(e^{jw}) \rangle^2} \quad (2.113)$$

Où on définit l'opération

$$\langle \beta(e^{jw}) \rangle = \frac{1}{K} \sum_{k=1}^K \beta^k(e^{jw}).$$

2.6 Les performances des algorithmes

L'objectif du réseau de microphones est d'améliorer le rapport signal/bruit (SNR) en ajoutant les signaux de manière cohérente et le bruit de manière incohérente. L'amélioration est mesurée par le gain du réseau.

Soit le réseau de N microphones, on note,

$$x_i(k) = s(k)a_i + n_i(k) \quad i = 1, \dots, N \quad (2.114)$$

Où

$x_i(k)$: le signal d'entrée à l' i ème microphone.

$s(k)$: le signal de la source.

$n_i(k)$: le bruit à l' i ème microphone.

$a_i = e^{j\omega\tau_i}$ est un déphasage.

On note aussi

$$\begin{aligned} \mathbf{x}(k) &= \begin{bmatrix} x_1(k) & \cdots & x_N(k) \end{bmatrix} \\ \mathbf{n}(k) &= \begin{bmatrix} n_1(k) & \cdots & n_N(k) \end{bmatrix} \\ \mathbf{a} &= \begin{bmatrix} a_1 & \cdots & a_N \end{bmatrix} \end{aligned}$$

donc

$$\mathbf{x}(k) = s(k)\mathbf{a} + \mathbf{n}(k)$$

En appliquant le formateur de voies sur les signaux d'entrées, on aura en sortie

$$y = \mathbf{w}^H \mathbf{x}(k) = y_s(k) + y_n(k) \quad (2.115)$$

Où $y_s(k)$ et $y_n(k)$ sont respectivement la partie du signal désiré, et la partie du bruit, en sortie du réseau.

2.6.1 Le gain du réseau

le rapport signal sur bruit (SNR_e) à l'entrée de chaque microphone

$$SNR_e = \frac{\sigma_s^2}{\sigma_n^2} \quad (2.116)$$

Où σ_s^2 est la puissance du signal désiré $s(k)$, σ_n^2 est la puissance du bruit $n_i(k)$ (on suppose que la puissance d'entrée du bruit est la même pour chaque microphone, $\sigma_n^2 = \sigma_{n_i}^2$, et le signal désiré et le bruit sont dé-corrélés).

le rapport signal sur bruit (SNR_s) à la sortie du réseau de microphones

$$SNR_s = \frac{\sigma_{y_s}^2}{\sigma_{y_n}^2} \quad (2.117)$$

Où $\sigma_{y_s}^2$ et $\sigma_{y_n}^2$ sont respectivement la puissance du signal désiré, et la puissance du bruit, en sortie du réseau.

Donc le gain du réseau de microphones est défini comme

$$\mathbf{G} = \frac{SNR_s}{SNR_e} \quad (2.118)$$

dans les parties suivantes de cette section, on va étudier le gain pour chaque algorithme.

2.6.2 performance de MVDR

De l'équation (2.17), et avec $\mathbf{a} = \mathbf{a}(\theta)$, les poids du formateur de voies MVDR deviennent

$$\mathbf{w}^H = \frac{\mathbf{R}_{xx}^{-1} \mathbf{a}}{\mathbf{a}^H \mathbf{R}_{xx}^{-1} \mathbf{a}} \quad (2.119)$$

Puisque le signal et le bruit sont dé-corrélés

$$\mathbf{R}_{xx} = \sigma_s^2 \mathbf{a} \mathbf{a}^H + \mathbf{R}_{nn} \quad (2.120)$$

On peut déterminer l'inverse de \mathbf{R}_{xx} en utilisant l'identité de Woodbury

$$\left[\sigma_s^2 \mathbf{a} \mathbf{a}^H + \mathbf{R}_{nn} \right]^{-1} = \mathbf{R}_{nn}^{-1} - \frac{\mathbf{R}_{nn}^{-1} \mathbf{a} \mathbf{a}^H \mathbf{R}_{nn}^{-1}}{\sigma_s^{-2} + \mathbf{a}^H \mathbf{R}_{nn}^{-1} \mathbf{a}} \quad (2.121)$$

On utilise cette dernière dans (2.119) on trouve le formateur de voies MVDR en fonction de la matrice de covariance du bruit.

$$\mathbf{w} = \frac{\mathbf{R}_{nn}^{-1} \mathbf{a}}{\mathbf{a}^H \mathbf{R}_{nn}^{-1} \mathbf{a}} \quad (2.122)$$

Puisque le formateur de voies MVDR ne touche pas le signal désiré (distortionless),

$$\sigma_{y_s}^2 = \sigma_s^2$$

On a

$$\sigma_{y_n}^2 = \mathbf{w}^H E \left[\mathbf{n}(k) \mathbf{n}(k)^H \right] \mathbf{w} = \mathbf{w}^H \mathbf{R}_{nn} \mathbf{w} \quad (2.123)$$

En remplaçant (2.122) dans l'expression de $\sigma_{y_n}^2$ on trouve,

$$\sigma_{y_n}^2 = \left[\mathbf{a}^H \mathbf{R}_{nn}^{-1} \mathbf{a} \right]^{-1} \quad (2.124)$$

Donc

$$\mathbf{G} = \sigma_s^2 \left[\mathbf{a}^H \mathbf{R}_{nn}^{-1} \mathbf{a} \right] / \frac{\sigma_s^2}{\sigma_n^2} = \sigma_n^2 \left[\mathbf{a}^H \mathbf{R}_{nn}^{-1} \mathbf{a} \right] \quad (2.125)$$

Dans le cas d'un bruit blanc ($\mathbf{R}_{nn} = \sigma_n^2 \mathbf{I}$), donc $\mathbf{G} = N$.

2.6.3 performance de l'algorithme retard et somme

Maintenant, nous pouvons examiner les SNR d'entrée et de sortie du DS beamformer. Le SNR d'entrée par rapport au signal de référence est

$$SNR_e = \frac{\sigma_s^2}{\sigma_e^2}$$

Après le formateur de voies Delay-and-sum, $\mathbf{w}_{DS} = \frac{1}{N}\mathbf{a}^H$, la puissance de signal désiré en sortie est :

$$\sigma_{y_s}^2 = \sigma_s^2$$

et d'après (2.123)

$$\sigma_{y_n}^2 = \frac{1}{N^2}\mathbf{a}^H \mathbf{R}_{nn} \mathbf{a} \quad (2.126)$$

Dans le cas du bruit blanc

$$\sigma_{y_n}^2 = \frac{\sigma_n^2}{N}$$

Donc le gain sera

$$\mathbf{G} = N$$

Ce qui implique que la performance du filtre MVDR est la même pour le filtre Delay-and-Sum lorsque les signaux de bruit observés sur le réseau sont dé-corrélés et ont la même puissance. L'avantage du filtre MVDR est que ce formateur de voies adaptatif, peut s'adapter à l'environnement, pour une réduction maximale du bruit.

2.6.4 performance de LCMV

La puissance du signal en sortie est la même qu'en entrée (distortionless)

$$\sigma_{y_s}^2 = \sigma_s^2$$

La puissance du bruit en sortie est

$$\sigma_{y_n}^2 = \mathbf{w}^H \mathbf{R}_{nn} \mathbf{w} = \sigma_n^2 |\mathbf{w}^H \boldsymbol{\rho}_n \mathbf{w}|$$

Où $\boldsymbol{\rho}_n$ est la matrice de covariance normalisée du bruit. Donc le gain du réseau est

$$\mathbf{G} = \left[\mathbf{w}^H \boldsymbol{\rho}_n \mathbf{w} \right]^{-1}$$

En remplaçant (2.52) dans l'expression du gain, on trouve

$$\mathbf{G} = \frac{1}{\mathbf{g}^H \left[\mathbf{C}^H \boldsymbol{\rho}_n^{-1} \mathbf{C} \right]^{-1} \mathbf{g}}$$

2.7 Conclusion

Dans ce chapitre, on a discuté les différents formateurs de voies qui existent dans la littérature, et qui permettent de fonctionner pour un réseau de microphones. On a vu l'utilisation du filtre à retard numérique pour choisir la direction d'arrivée d'un signal large bande, comme le signal de la parole. On a présenté les formateurs de voies les plus utilisés pour le signal de la parole, et surtout les formateurs de voies à filtre à réponse impulsionnelle comme le formateur de voies de frost. Les formateurs de voies adaptatifs donnent le meilleur moyen pour avoir plus de performance et moins de calcul. On a terminé avec les performances des algorithmes et le gain de chacun d'eux.

Chapitre 3

Tests et résultats

Dans cette section, on fait les tests pour visualiser la performance de chaque algorithme. Dans la partie (3.1), on fait des simulation avec un réseau de microphones synthétiques, et en utilisant plusieurs sources. Dans la partie (3.2) on passe aux tests expérimentaux, où on utilise un réseau de microphones réel avec 4 microphones, dans un environnement réel.

3.1 Les résultats de simulation

Pour la simulation, on utilise 3 sources provenant de directions différentes, figure 3.1. On essaye d'extraire une seule source tout en éliminant les autres sources, on fait plusieurs tests en changeant les paramètres (direction d'arrivée, direction d'observation,...). Dans la simulation, et pour voir la différence entre les algorithmes, on fixe les directions d'arrivée des signaux à 0 degrés, 30 degrés et -30 degrés.

On visualise la performance des algorithmes en mesurant le gain du réseau, et en comparant le signal de sortie avec le signal désiré.

3.1.0.1 Définir le réseau de microphones

On fait la simulation en utilisant un signal et un réseau de microphones synthétiques. Pour définir un signal de source arrivant au réseau de la direction θ , on crée le signal \mathbf{x}_n , $n = 1, \dots, N$ arrivant au n ème microphone du réseau en dupliquant le signal de la source (qui est un enregistrement) avec un retard $\tau_n = \frac{nd \sin \theta}{c}$ secondes entre le microphone de référence (qui est le premier microphone) et le n ème microphone, où d (m) est la distance entre 2 microphones adjacents, c (m/s) est la vitesse du son. Pour changer la distance d entre 2 microphones adjacents, ou la direction d'arrivée, il suffit juste d'utiliser l'expression du retard τ_n .

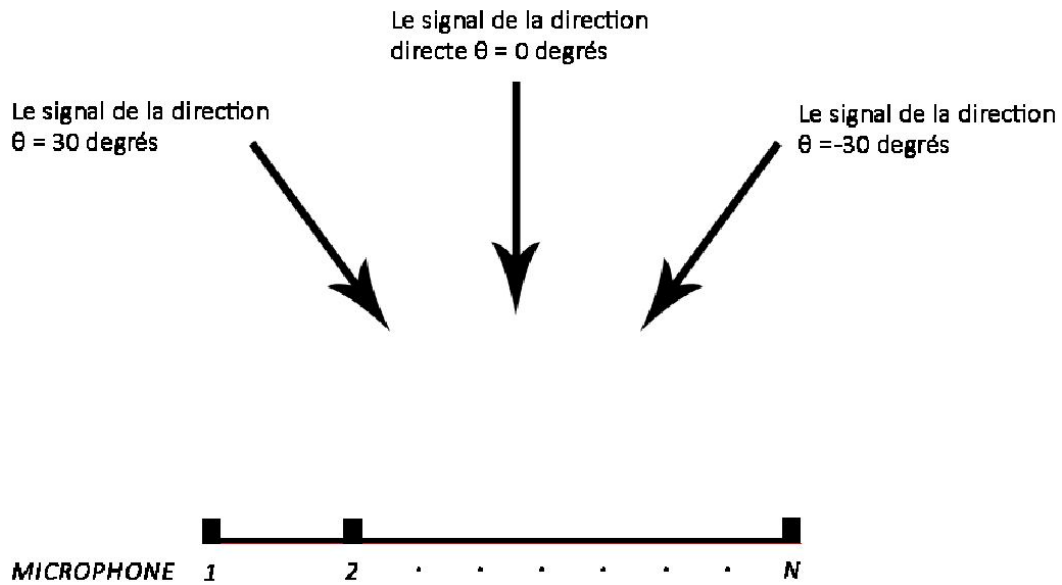


FIGURE 3.1 – Illustration de l’environnement de simulation, le réseau de microphones reçoit 3 sources de 3 directions différentes : $\theta = 0$ degrés, $\theta = 30$ degrés, $\theta = -30$ degrés.

3.1.0.2 Mesurer les performances

La mesure de la performance est un peu simple par rapport aux mesures expérimentales. On utilise 3 sources (figure 3.1). Le gain du réseau de microphones, comme il a été illustré précédemment dans la section (2.6) est

$$G = \frac{SNR_s}{SNR_e}$$

Et on sait que les algorithmes laissent le signal désiré non distordu, donc la puissance du signal, σ_s^2 reste la même en sortie ($\sigma_s^2 = \sigma_{y_s}^2$), on déduit

$$G = \frac{\sigma_n^2}{\sigma_{y_n}^2}$$

Où σ_n^2 est la puissance du bruit en entrée du réseau (dans notre cas le bruit est les deux autres interférences), et $\sigma_{y_n}^2$ est la puissance du bruit en sortie du réseau,

$$G_{dB} = 10 \log \frac{|\mathbf{i}_1 + \mathbf{i}_2|^2}{|\mathbf{y}_s - \mathbf{s}|^2}$$

Où, \mathbf{i}_1 , \mathbf{i}_2 , \mathbf{y}_s et \mathbf{s} sont respectivement, la 1ère interférence, la 2ème interférence, la sortie du formateur de voies, et le signal désiré.

3.1.1 La performance de l’algorithme Delay-and-sum

En appliquant le formateur de voies Delay-and-Sum, avec $N = 4$ microphones, et angle d’observation $\theta = 0$ degrés, la figure 3.2 illustre le signal d’entrée (le signal du premier microphone), le signal de sortie, et le signal désiré.

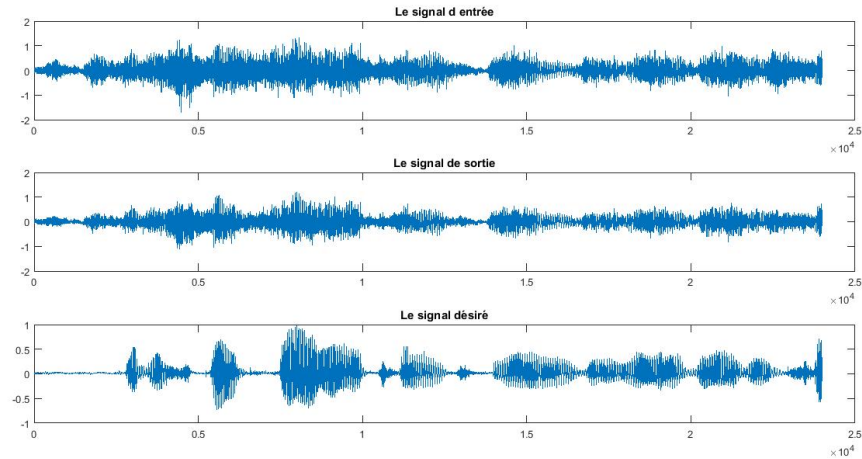


FIGURE 3.2 – le signal d’entrée, le signal de sortie et le signal désiré du formateur de voies Delay-and-sum avec $N = 4$ microphones.

De la figure, on voit que le signal de sortie ressemble plus au signal d’entrée, donc le traitement n’a pas réussi à isoler le signal utile. Ce résultat n’était pas seulement visualiser, on a entendu la sortie et on l’a comparé avec l’entrée, il y a une petite amélioration mais il y a aussi une forte présence de bruit. Avec $N = 10$ microphones, la performance augmente, et la ressemblance entre le signal de sortie et le signal désiré est très remarquable, (Figure 3.3).

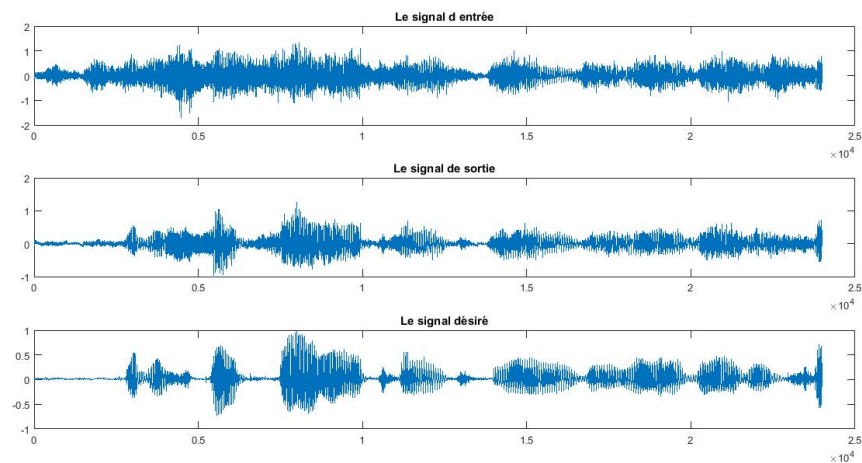


FIGURE 3.3 – le signal d’entrée, le signal de sortie et le signal désiré du formateur de voies Delay-and-sum avec $N = 10$ microphones.

Le filtrage spatial avec le formateur de voies Delay-and-Sum ne donne pas un très bon résultat, surtout avec un faible nombre de microphones. Le filtre spatial Delay-and-Sum a augmenté le rapport signal sur bruit SNR en sortie.

Avec $N = 4$ microphones le gain est $G_{dB} = 3.8$ dB, et avec 10 microphones le gain est $G_{dB} = 6.9$ dB, l'augmentation du nombre de microphones, augmente le gain.

3.1.2 La performance de l'algorithme MVDR dans le domaine fréquentiel

L'algorithme MVDR dans le domaine fréquentiel donne une bonne performance par rapport à l'algorithme Delay-and-Sum, avec 4 microphones, cet algorithme nous donne le résultat dans la Figure 3.4. Avec 4 microphones le gain du réseau est $G_{dB} = 6.6$ dB.

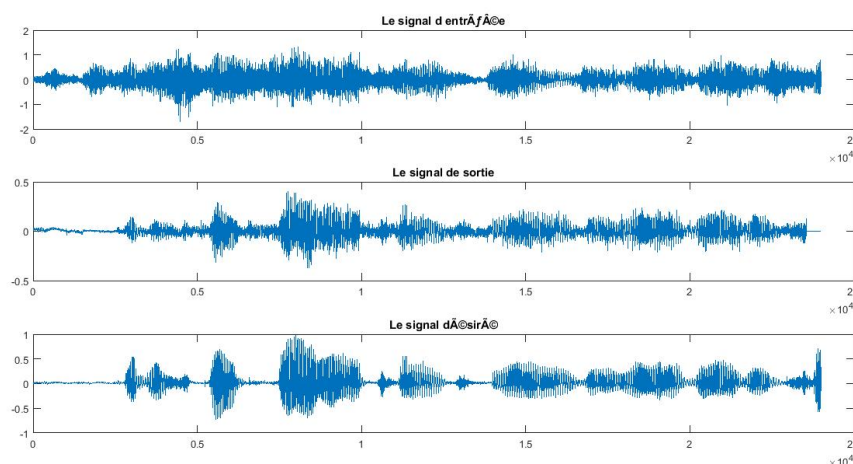


FIGURE 3.4 – le signal d'entrée, le signal de sortie et le signal désiré du formateur de voies MVDR fréquentiel avec $N = 4$ microphones.

3.1.3 La performance de l'algorithme de frost

L'algorithme de frost est peut-être parmi les algorithmes les plus robustes. Cet algorithme garde une réponse fixe dans la direction du signal désiré en supprimant les signaux provenant d'autres directions. On constate que le SNR en sortie est devenu meilleur par rapport à l'algorithme Delay-and-Sum.

Avec 4 microphones, et un angle d'observation $\theta = 0$ degrés l'algorithme donne un très bon résultat par rapport les autres algorithmes, figure 3.5.

Avec 10 microphones, on obtient plus de performance, et on ne peut pas observer la différence entre le signal de sortie et le signal désiré, figure 3.6.

Toutes les versions de l'algorithme de Frost (direct, LMS, GSC, GSC adaptatif) ont le même résultat en simulation, des Figures 3.5 et 3.6.

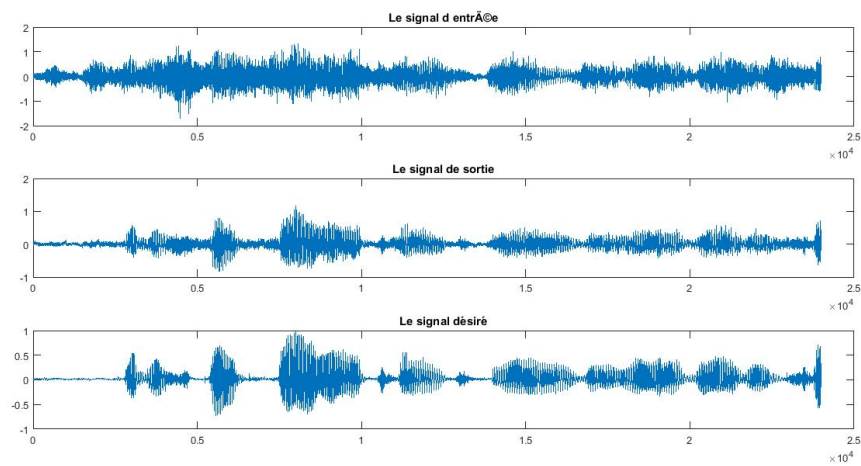


FIGURE 3.5 – le signal d'entr e, le signal de sortie et le signal d sir  du formateur de voies Frost avec $N = 4$ microphones.

la performance de cet algorithme est vraiment int ressante, et elle a d pass  la performance des autres algorithmes, avec 4 microphones seulement $G_{dB} = 9.6$ dB, avec 10 microphones $G_{dB} = 11.5$ dB.. Le formateur de voies de Frost entre tous les formateur de voies test s, a donn  un tr s bon r sultat.

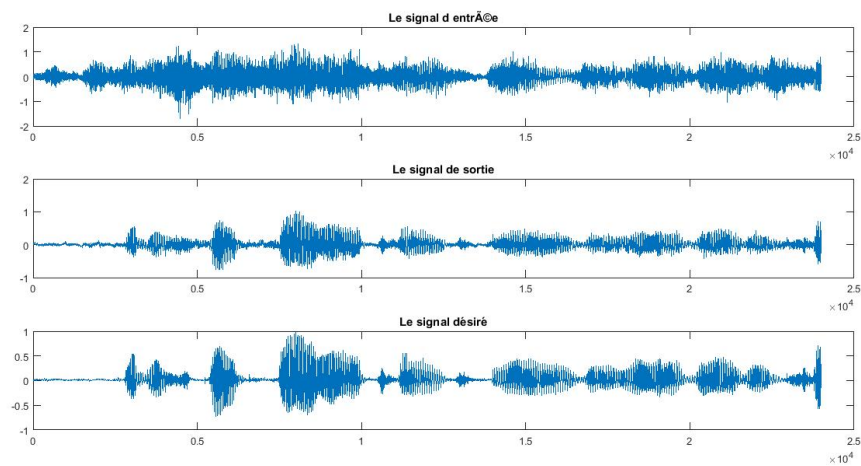


FIGURE 3.6 – le signal d'entr e, le signal de sortie et le signal d sir  du formateur de voies Frost avec $N = 10$ microphones.

3.2 Les résultats expérimentaux

Dans cette partie, on présente les tests expérimentaux, en utilisant un réseau de microphones réel de 4 microphones et dans un environnement réel.

On réalise un réseau de 4 microphones, pour faire des tests et des mesures. On teste plusieurs algorithmes avec ce réseau et on analyse les résultats et les performances de chaque algorithme.

On place le réseau de microphones dans une salle de taille $5m$ de longueur, $4m$ de largeur, $4m$ de hauteur ($x \times y \times z$), figure 3.7. La première source de la parole est dans la direction directe du réseau ($\theta = 0$ degrés), et la deuxième source de la parole qui représente l'interférence est dans une autre direction ($\theta = 30$ degrés). On ajoute aussi un bruit blanc dans chaque microphone.

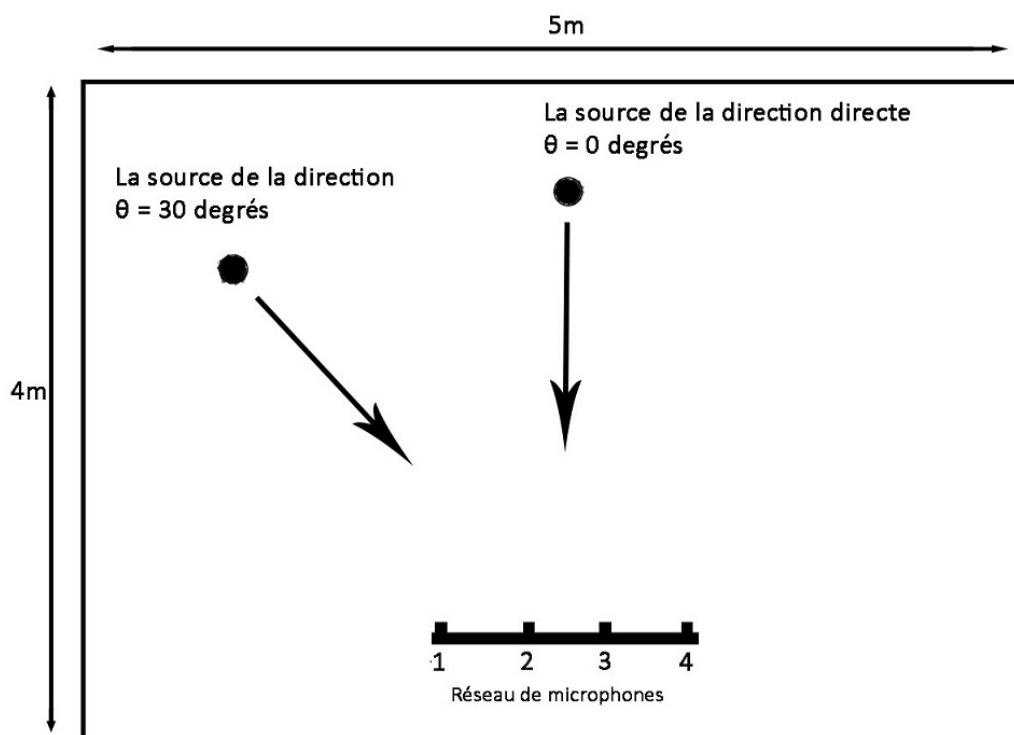


FIGURE 3.7 – l'environnement des expériences, le réseau de microphones reçoit : une source de la direction $\theta = 0$ degrés, une source de la direction $\theta = 30$ degrés.

3.2.1 Le montage des composants

Pour tester les algorithmes de formations de voies, avec un signal réel, on a besoin d'acquérir le signal en utilisant une carte d'acquisition, la Figure (3.8) montre le montage du système d'acquisition.

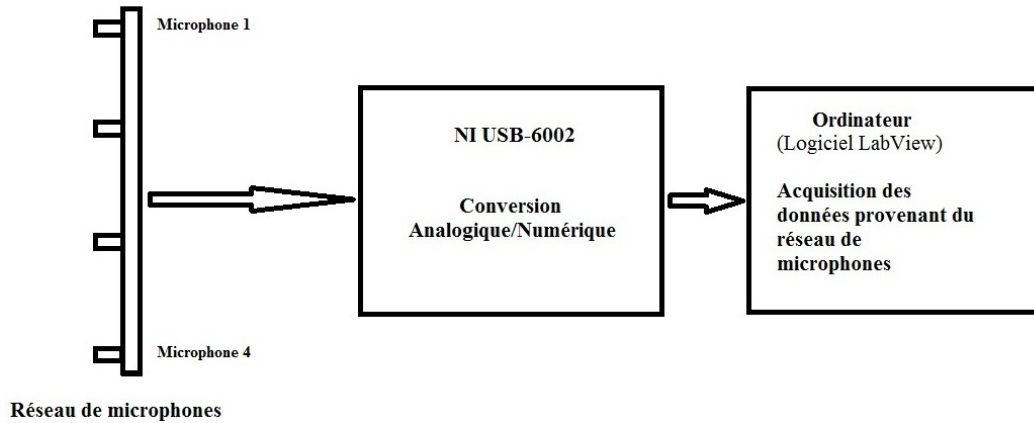


FIGURE 3.8 – Schéma bloc du montage du système d'acquisition.

Le réseau de microphones

On réalise un réseau de microphones linéaire en utilisant 4 microphones, la Figure 3.9 montre le réseau de microphones utilisé. Avec une distance $d = 5\text{cm}$ entre deux microphones adjacents. La distance d est choisi pour éviter le problème d'aliasing (chapitre 1). On prend $d = \frac{\lambda}{2}$, où λ est la longueur d'onde de la plus grande fréquence du signal de la parole, pour $d = 5\text{cm}$, donc $\lambda = 10\text{cm}$ qui correspond à la longueur d'onde de la fréquence $f_{max} = \frac{c}{\lambda} = \frac{340}{0.1} = 3400\text{Hz}$.

On a utilisé le microphone de référence KS-003 qui a 3 pins, les 2 pins (Vcc, Ground) sont pour l'alimentation (qui doit être entre 6V et 14V), le 3ème pin est la sortie du microphone qui est un signal analogique.

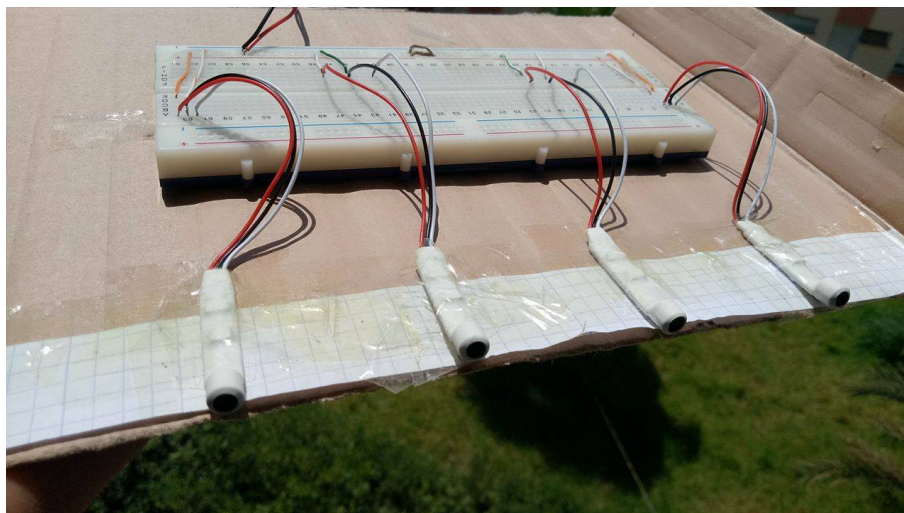


FIGURE 3.9 – Le réseau de microphones réalisé, avec $N = 4$ microphones et $d = 5\text{ cm}$.

3.2.1.1 La carte d'acquisition NI USB-6002

Les signaux provenant des microphones sont des signaux analogiques, pour pouvoir les traiter et stocker dans un ordinateur on doit les transformer en signaux numériques. La carte d'acquisition NI USB-6002 [18] (Figure 3.10) permet de faire l'acquisition des données du réseau de microphones. La carte réalise la conversion Analogique/Numérique des signaux provenant des microphones.

La carte d'acquisition NI USB-6002 contient 8 entrées analogique/numérique. La fréquence d'échantillonnage maximale est $50kHz$ (c'est-à-dire si on utilise seulement une seule entrée analogique la fréquence d'échantillonnage maximale sera $50kHz$, mais si on utilise 2 entrées analogiques la fréquence d'échantillonnage maximale diminue à $25kHz$, et ainsi de suite pour plusieurs entrées analogiques), dans notre cas, on a utilisé 4 entrées analogiques (4 microphones), donc la fréquence d'échantillonnage maximale est $12.5kHz$.

Le constructeur de cette carte, National Instrument, fournit un environnement de développement très avancée pour faciliter son utilisation, et faire l'acquisition de données avec un simple programme.

Avec le logiciel LABVIEW, on peut configurer la carte NI USB-6002 facilement, on peut enregistrer les données acquies dans un fichier pour pouvoir les utiliser et les tester avec des algorithmes sur MATLAB.



FIGURE 3.10 – La carte NI USB-6002.

La carte a un seul ADC (Convertisseur Analogique/Numérique), donc l'échantillonnage se fait de façon multiplexée, alors que nous avons besoin de plusieurs ADC pour faire l'échantillonnage simultané, pour

prendre un vecteur d'échantillons dans le même instant. D'après les tests qu'on a fait avec le réseau, où on a mis une source de signal dans la direction direct du réseau (c'est-à-dire, il n'y aura pas de décalage entre les signaux de microphones), on a observé que tous les signaux du réseaux sont alignés, et il n'y avait pas de décalage, donc cet échantillonnage ne cause pas de problème.

3.2.1.2 Configurer le système d'acquisition

les étapes de configuration du système d'acquisition sont :

1. Installer le logiciel LABVIEW.
2. Installer le driver de la carte d'acquisition du CD de la carte NI USB-6002.
3. Brancher les microphones du réseau sur les entrées analogiques de la carte NI USB-6002
4. Suivre [19] pour faire le programme sur le logiciel LABVIEW.

3.2.2 Le test du système d'acquisition

Après la configuration du système d'acquisition, on peut tester leur fonctionnement. On attaque le réseau de microphones par une source sonore à bande étroite, pour bien visualiser le décalage entre les signaux du réseau, la fréquence d'échantillonnage des signaux est 12500Hz . La distance entre la source et le centre du réseau de microphones est 1m.

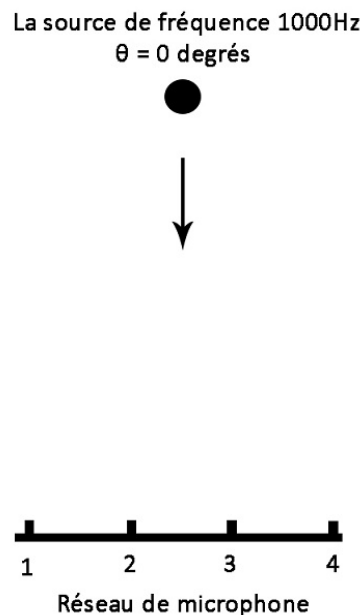


FIGURE 3.11 – La source est dans la direction $\theta = 0$ degrés, la distance entre la source et le centre du réseau est 1m.

Dans le premier test, on met une source sonore de fréquence 1000Hz dans la direction directe au réseau ($\theta = 0$ degrés), figure 3.11. La figure 3.12 montre une portion des 4 signaux du réseau.

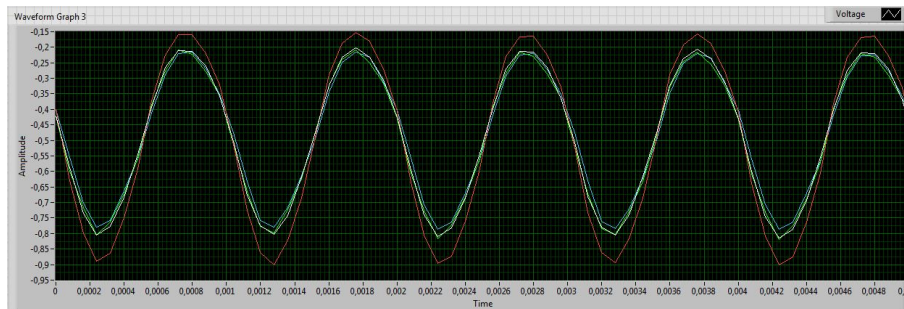


FIGURE 3.12 – Les signaux du réseau de microphones réalisés avec 4 microphones, la source est une source monochromatique ($f=1000\text{Hz}$), la fréquence d'échantillonnage est 12500Hz , $\theta = 90^\circ$.

On voit que les quatre signaux sont en phase, mais leurs amplitudes sont différentes l'un par rapport aux autres, à cause de la différence des canaux entre la source et chaque microphone du réseau. En calculant le décalage τ qu'il doit être entre deux microphones adjacents, on trouve $\tau = \frac{d \sin \theta}{c} = 0\text{s}$.

Dans le deuxième test, on met la source sonore de fréquence 1000Hz dans la direction $\theta = 90^\circ$ du réseau, la figure 3.13. La figure 3.14 montre une portion des 4 signaux du réseau.



FIGURE 3.13 – La source est dans la direction $\theta = 90^\circ$, la distance entre la source et le centre du réseau est 1m.

De la figure 3.14, on lie chaque signal à son microphone correspondant :

- Le signal vert correspond au microphone 1.
- Le signal blue correspond au microphone 2.
- Le signal rouge correspond au microphone 3.
- Le signal blanc correspond au microphone 4.

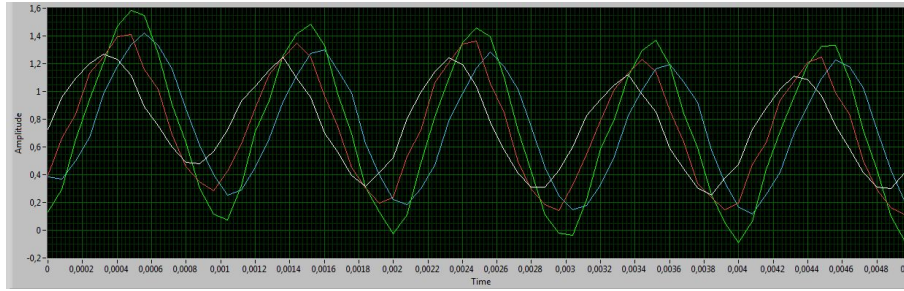


FIGURE 3.14 – Les signaux du réseau de microphones réslisé avec 4 microphones, la source est une source monochromatique ($f=1000\text{Hz}$), la fréquence d'échantillonnage est 12500Hz , $\theta = 90$ degrés.

On voit que les quatre signaux sont décalés, et leurs amplitudes sont différents l'un par rapport aux autres, à cause du canal entre la source et chaque microphone du réseau. Théoriquement, le décalage entre deux signaux de microphones adjacents est $\tau = \frac{d \sin \theta}{c} = \frac{0.05 \sin 90}{340} = 0.15\text{ms}$, En mesurant le décalage dans la figure 3.14, on trouve le même résultat.

3.2.3 Les résultats expérimentaux des algorithmes

Nous avons utilisé seulement 4 microphones. les performances obtenues sont observables, et il suffit juste d'augmenter le nombre de microphones pour avoir plus de performance.

En utilisant les données acquises du réseau de microphones (figure 3.9), on a fait les tests de plusieurs algorithmes, la performance de chacun d'eux est différente de ce que nous avons obtenu dans la simulation. On va montrer la puissance et la robustesse des algorithmes adaptatifs.

Dans cette section, on visualise les performances de chaque algorithme, et on va voir que les algorithmes adaptatifs résulte d'une meilleure performance par rapport aux autres algorithmes non adaptatifs. On doit prendre en considération que le nombre de microphones utilisé dans le réseau est très important, on a vu dans la partie simulation que l'augmentation du nombre de microphones a augmenté le gain, ceci est aussi valide avec les signaux réels, donc pour avoir de bons résultats il faut utiliser un grand nombre de microphones, parce qu'avec le signal de la parole où la réverbération est très élevée et la bande passante est large, il est vraiment difficile de faire le beamforming. Dans un projet de MIT [20] les chercheurs ont utilisé 1020 microphones [21], (Figure 1.3), et aussi pour les réseaux de microphones commercialisés, où ils ont utilisé plus de 20 microphones pour avoir une performance remarquable, figure 1.4 et figure 1.5.

3.2.3.1 L'algorithme Delay-and-Sum

La Figure 3.15, montre que cet algorithme a filtré les autres directions et le bruit blanc. L'algorithme Delay-and-Sum est un algorithme fixe (Fixed algorithm), donc il est indépendant du signal d'entrée. On déduit que cet algorithme, malgré sa simplicité, il donne une performance acceptable.

L'algorithme Delay-and-Sum n'a pas donné un très bon résultat en simulation par rapport aux autres algorithmes, mais avec un signal réel, on peut dire qu'il a donné une bonne performance.

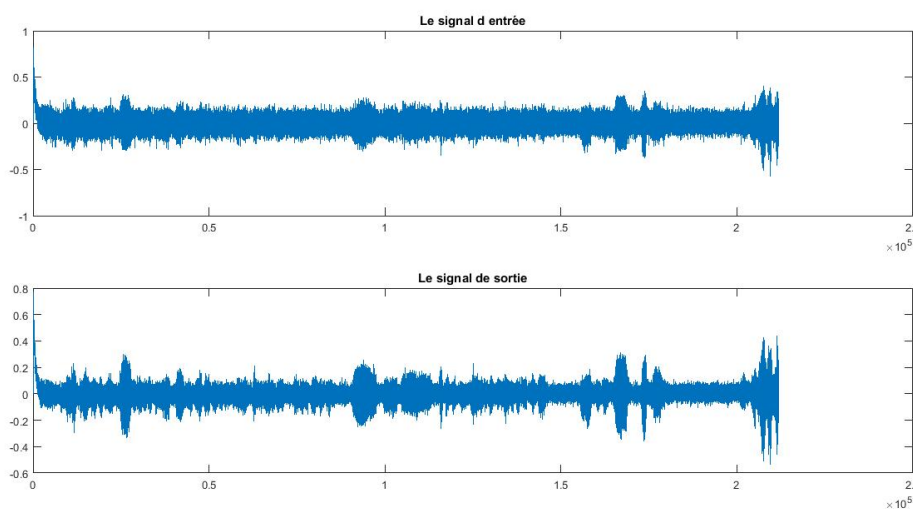


FIGURE 3.15 – Le signal d'entrée et le signal de sortie avec l'algorithme Delay and Sum $N = 4$.

3.2.3.2 L'algorithme LCMV avec la version directe

Les tests avec l'algorithme de Frost avec la version direct, en utilisant un signal réel, ne donne pas un résultat intéressant (Figure 3.16), et il n'a pas augmenté le gain du réseau. La réverbération et les trajets multiples empêchent l'algorithme à mieux faire le filtrage. Dans la version directe, où il faut estimer la matrice de covariance des signaux d'entrée, l'algorithme n'a pas pu mettre à jour la matrice de covariance pour confronter le changement de l'environnement avec la réverbération, l'atténuation et les trajets multiples.

3.2.3.3 L'algorithme de Frost LMS

Le formateur de voies de Frost avec la version adaptative LMS, et avec seulement 4 microphones, on obtient un résultat observable. L'algorithme adaptatif proposé par Frost en 1972, a la capacité de s'adapter au changement de l'environnement, à la réverbération et les trajets multiples. Cet algorithme a pu se confronter aux conditions du signal réel, il donne un bon résultat. Dans la figure 3.17, et en comparant entre le signal d'entrée et le signal de sortie on voit que l'algorithme a donné une bonne performance.

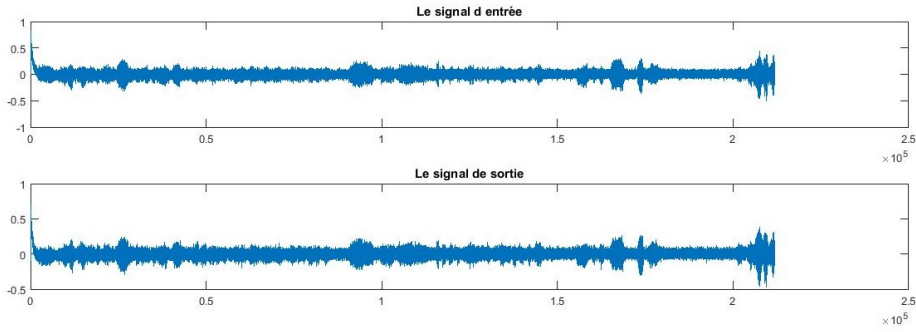


FIGURE 3.16 – Le signal d’entrée et le signal de sortie avec l’algorithme Frost direct $N = 4$.

L’algorithme adaptatif avec l’implémentation LMS diminue la complexité de calcul. L’exécution de cet algorithme est plus rapide en la comparant avec l’implémentation direct.

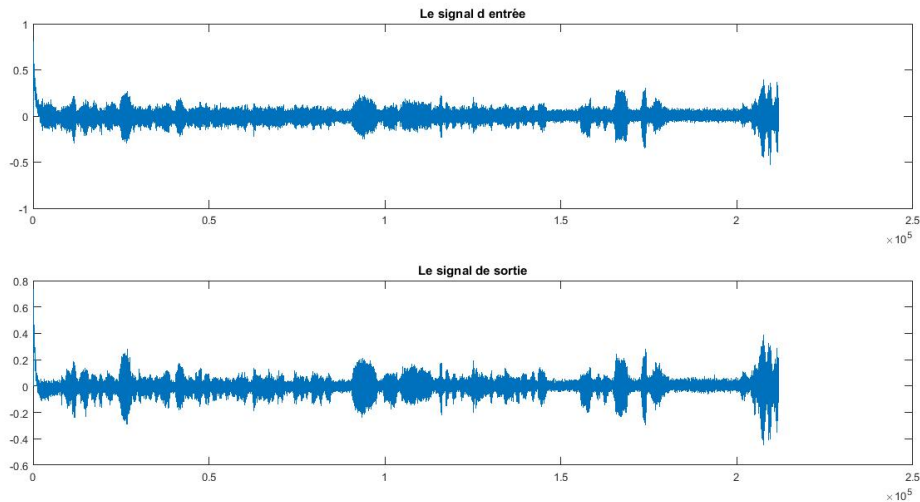


FIGURE 3.17 – Le signal d’entrée et le signal de sortie avec l’algorithme Frost LMS $N = 4$.

3.2.3.4 L’algorithme de Frost LMS avec la structure GSC

Avec la structure Generalized Sidelobe Canceller (GSC), l’algorithme de Frost est devenu plus intéressant, la structure GSC, a donné plus de réduction de bruit. La figure 3.18, montre la performance de cet algorithme. On observe moins de bruit dans le signal de sortie par rapport au signal d’entrée.

L’implémentation direct de l’algorithme de frost, n’a pas pu fonctionner avec un signal réel. Le signal de la parole réel est un signal non stationnaire, et avec l’implémentation directe, l’algorithme ne s’adapte pas au signal, alors qu’avec un algorithme adaptatif comme l’algorithme de Frost avec la version LMS, le

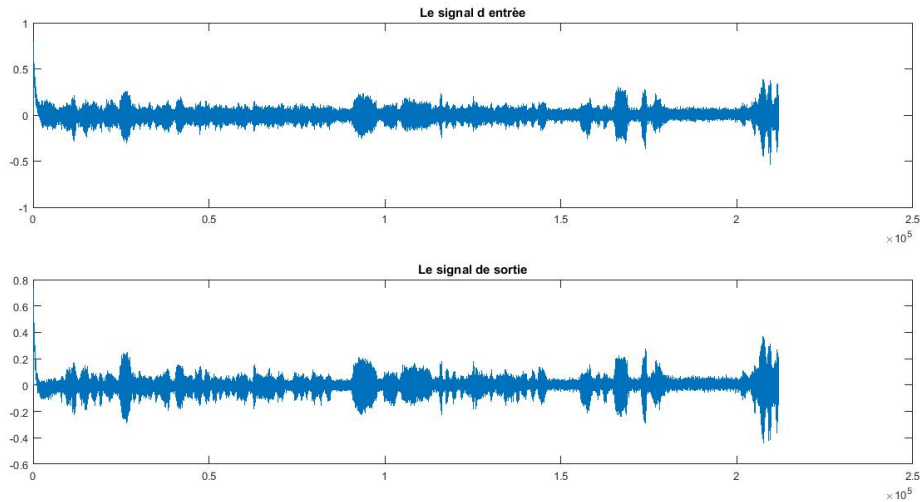


FIGURE 3.18 – Le signal d’entrée et le signal de sortie avec l’algorithme Frost LMS et avec la structure GSC $N = 4$.

formateur de voies s’adapte avec chaque échantillon, donc ces algorithmes donnent le meilleur résultat possible.

les algorithmes adaptatifs (implémentation LMS, implémentation GSC LMS) offrent la possibilité de réaliser un réseau de microphones qui peut fonctionner en temps réel. En voyant les algorithmes adaptatifs on remarque qu’un processeur peut exécuter toutes les instructions de l’algorithme dans une durée beaucoup moins par rapport un algorithme à implémentation directe. Une implémentation directe (l’implémentation directe de l’algorithme de frost) pour un réseau de 10 microphones, et un filtre FIR de taille 20, la matrice de covariance \mathbf{R}_{xx} est de taille 200×200 . L’implémentation directe exige l’inversion de la matrice de covariance, cette opération prend beaucoup de temps de calcul, et empêche le système de fonctionner en temps réel.

3.3 Conclusion

Dans ce chapitre, on a visualisé les performances des formateurs de voies. Dans la simulation, on a vu la performance des algorithmes avec un réseau de microphones synthétiques, et aussi une comparaison entre les différents algorithmes. Dans l’expérimentation, on a réalisé le réseau de 4 microphones, et on a testé les différents algorithmes. On a constaté que les algorithmes adaptatifs comme l’algorithme de Frost avec la version LMS, donnent le meilleur résultat par rapport aux autres algorithmes.

Conclusion

L'application la plus importante envisagée dans ce mémoire concerne l'extraction d'une source mélangée avec les interférences et le bruit, en connaissant juste sa position.

Les résultats montrent qu'avec un nombre suffisant de microphones on réussit à faire ce filtrage. On a trouvé des algorithmes de filtrage spatial qui peuvent fonctionner avec un signal large bande, comme le signal de la parole. Les formateurs de voies à filtre à réponse impulsionnelle finie (FIR), comme le formateur de voies de Frost, ont prouvé une bonne robustesse et ils ont donné des bons résultats, tandis que les formateurs de voies classiques n'ont pas réussi à extraire le signal utile. Les algorithmes adaptatifs, comme LMS, ont donné plus de performance et moins de temps de calcul, et ils sont l'unique choix si on voudrait fonctionner le réseau de microphones en temps réel.

On a testé les algorithmes avec des données synthétiques ainsi qu'avec des données réelles enregistrées par notre réseau. Les tests de simulation ont visualisé les performances des algorithmes, on a remarqué que l'augmentation du nombre de microphones permet d'améliorer la performance. Les algorithmes ont filtré les interférences et le bruit des autres directions, et on a pu entendre le signal utile. Les tests expérimentaux nous ont permis d'appliquer les algorithmes sur un signal réel, où l'environnement est réverbérant et il n'y a pas que le signal de la direction directe. D'après les résultats expérimentaux, on a observé que la performance des algorithmes avec un signal réel est un peu différente des résultats de simulation, car la matrice A qui produit les signaux synthétiques ne tient pas en compte des conditions de l'environnement. Donc pour réussir à faire le filtrage spatial pour les sources sonores il faut bien calibrer le réseau suivant l'environnement et avoir un nombre suffisant de microphones, 8 à la rigueur. Si on a une bonne caractérisation de la salle, on peut l'inclure dans les contraintes.

Bibliographie

- [1] W. Liu and S. Weiss. *Wireless Communications and Mobile Computing : Wideband Beamforming : Concepts and Techniques*. John Wiley and Sons, Ltd, USA, 2010.
- [2] H. Krim and M. Viberg. Two decades of array signal processing research : the parametric approach. *IEEE Signal Processing Magazine*, 13(4) :pp. 67–94, july 1996.
- [3] F. Alton Everest. *Master Handbook of Acoustics*. McGraw-Hill, New York, 2009.
- [4] M. Woelfel and J. McDonough. *Distant Speech Recognition*. Wiley, Hoboken, NJ,USA, 2009.
- [5] B. Owsinski. *Recording Engineer’s Handbook (2nd Edition)*. Course Technology PTR, Boston, MA, USA, 2009.
- [6] J. Benesty, J. Chen and Y. Huang. *Microphone array signal processing*, volume 1. Springer, Montréal, QC/ Murray Hill, NJ/ Bridgewater, NJ, 2008.
- [7] B.D. Van Veen and K.M. Buckley. Beamforming : a versatile approach to spatial filtering. *IEEE ASSP Magazine*, 5(2) :pp. 4–24, April 1988.
- [8] M. Brandstein and D. Ward, editors. *Microphone Arrays Signal Processing : Techniques and Applications*. springer, Berlin Heidelberg, 2001.
- [9] V. Valimaki T.I. Laakso and M. Karjalainen. Splitting the unit delay [fir/all pass filters design]. *IEEE Signal Processing Magazine*, 13(1) :pp. 30–60, Jan 1996.
- [10] H. L. Van Trees. *Optimum Array Processing. Part IV of Detection, Estimation, and Modulation Theory*. John Wiley and Sons, Inc, New York, 2002.
- [11] R.C. Bilger C.R. Lansing W.D.J. O’Brien B.C. Wheeler M.E. Lockwood, D.L. Jones and A.S. Feng. Performance of time- and frequency-domain binaural beamformers based on recorded signals from real rooms. *The Journal of the Acoustical Society of America*, 115 :pp. 379–391, 2004.
- [12] O. L. Frost. An algorithm for linearly constrained adaptive array processing. In *Proceedings of the IEEE*, volume Vol. 60, pages 926 – 935. IEEE, August 1972.
- [13] J. Bourgeois and W. Minker (eds.). *Time-Domain Beamforming and Blind Source Separation : Speech Input in the Car Environment*, volume 3 of *Lecture Notes in Electrical Engineering 3*. Springer US, first edition, 2009.
- [14] L. J. Griffiths and C. W. Jim. An alternative approach to linearly constrained adaptive beamforming. *IEEE Transactions on Antennas and Propagation*, 30 :27–34, January 1982.

- [15] W. Liu Y. Zhao and R. Langley. Application of the least squares approach to fixed beamformer design with frequency-invariant constraints. *IET Signal Process*, 5(3) :pp. 281–291, 2011.
- [16] D. Burshtein Sh. Gannot and E. Weinstein. Signal enhancement using beamforming and nonstationarity with applications to speech. *IEEE TRANSACTIONS ON SIGNAL PROCESSING*, 49(8) :pp. 1614–1626, August 2001.
- [17] O. Shalvi and E. Weinstein. System identification using nonstationary signals. *IEEE TRANSACTIONS ON SIGNAL PROCESSING*, 44(8) :pp. 2055–2063, August 1996.
- [18] Ni usb-6001/6002/6003 user guide - national instruments. <http://www.ni.com/pdf/manuals/374259a.pdf>.
- [19] Daq labview tutorial. <https://www.youtube.com/watch?v=Yi3SGjk3WSs>.
- [20] A. Agarwal E. Weinstein, K. Steele and J. Glass. Loud : A 1020-node modular microphone array and beamformer for intelligent computing spaces. MIT Computer Science and Artificial Intelligence Laboratory, 2004.
- [21] Loud array switching between multiple speakers talking at the same time. <https://www.youtube.com/watch?v=gGlrY46nfe4>.