Titre: Title:	Une méthode linéaire de déréverbération pour les signaux de parole
Auteur: Author:	Semah Aissaoui
Date:	2021
Type:	Mémoire ou thèse / Dissertation or Thesis
Référence: Citation:	Aissaoui, S. (2021). Une méthode linéaire de déréverbération pour les signaux de parole [Mémoire de maîtrise, Polytechnique Montréal]. PolyPublie. https://publications.polymtl.ca/6337/

POLYTECHNIQUE MONTRÉAL
affiliée à l'Université de Montréal

Une méthode linéaire de déréverbération pour les signaux de parole

SEMAH AISSAOUI
Département de mathématiques et de génie industriel

Mémoire présenté en vue de l'obtention du diplôme de *Maîtrise ès sciences appliquées*
Mathématiques appliquées

Mai 2021

POLYTECHNIQUE MONTRÉAL
affiliée à l'Université de Montréal

Ce mémoire intitulé :

Une méthode linéaire de déréverbération pour les signaux de parole

présenté par **Semah AISSAOUI**
en vue de l'obtention du diplôme de *Maîtrise ès sciences appliquées*
a été dûment accepté par le jury d'examen constitué de :

Michel GENDREAU, président
Antoine SAUCIER, membre et directeur de recherche
Richard LABIB, membre

DÉDICACE

À mes parents qui m'ont toujours supporté, mon cher père Mohsen et ma chère mère Hayet : je n'arriverai jamais à rendre ce qu'ils m'ont donné, ils sont la lumière de ma vie.

À mes frères Souhail, Oussama et mon adorable sœur Sirine,
que je suis fier de les avoir dans ma vie.

À mon oncle Ezzeddine, que je remercie pour tous ses conseils précieux.

À mon âme sœur, Rim qui n'a jamais cessé de m'encourager tout au long de ma maitrise.

À mes amis, Rami, Adnen, Yosri et Aymen qui ont été toujours à mes côtés.

Enfin, aux gens qu'on a perdus à cause du COVID-19, paix soit en leur âme....

REMERCIEMENTS

Je tiens à remercier en premier lieu mon directeur de recherche, Antoine Saucier, pour avoir accepté de m'encadrer pendant ma maitrise, pour sa disponibilité, son aide et ses conseils très précieux tout au long du travail ainsi qu'au cours de la rédaction de ce mémoire.
Je remercie également Wissem Maazoun pour l'effort qu'il a fait dans la démarche de ce projet de recherche. Je remercie aussi fluent.ai et Mitacs pour avoir accepté de financer ce projet de recherche.
En outre, je tiens à remercier Vikrant Singh Tomar, de m'avoir accueilli au sein de son organisation fluent.ai. Je remercie aussi toute l'équipe de fluent.ai et spécialement mon encadrant Hanwook Chung avec qui j'ai eu l'occasion de collaborer pendant mes activités de recherche.
J'adresse mes remerciements aussi à Michel Gendreau et Richard Labib qui ont accepté de faire partie de mon jury.
Enfin, je souhaite exprimer toute ma gratitude à mes parents. Je suis très content de les avoir dans ma vie. Merci d'avoir cru en moi. Je vous aime.

RÉSUMÉ

Lorsqu'une source sonore émet un signal audio dans une salle fermée et si le microphone est situé à une certaine distance de la source, le signal capté par le microphone est composé de plusieurs versions retardées et atténuées du signal source. En effet, le signal observé est formé du signal direct de la source et de ses réflexions avec les objets existants dans la même salle. Ce phénomène s'appelle la réverbération. Plusieurs études ont montré que la réverbération provoque la dégradation de la qualité du signal de parole et peut nuire à son intelligibilité. Pour cela, notre partenaire fluent.ai cherche une solution de déréverbération pour améliorer la performance de son système de reconnaissance automatique de la parole. Le problème de déréverbération est plus compliqué que le problème de réduction de bruit de fond car les réverbérations qu'on veut réduire sont en forte corrélation avec le signal désiré qu'on veut estimer.

C'est dans ce cadre qu'on effectue ce projet de recherche, notre objectif est d'étudier l'effet et les caractéristiques de la réverbération afin d'implémenter une solution qui permet de la réduire. Puisque la solution de déréverbération sera intégrée dans un système de reconnaissance automatique de la parole, on cherche une méthode qui peut fonctionner en temps réel et qui utilise le signal capté par un ou plusieurs microphones.

On commence par étudier la réverbération, ses caractéristiques et ses effets sur les signaux de parole. On présente aussi un état de l'art des techniques les plus connues de déréverbération. Parmi les méthodes de déréverbération présentées, la méthode de déréverbération par prédiction linéaire sera choisie et expliquée.

On présentera la méthode de déréverbération par prédiction linéaire avec le modèle proposé dans [1]. Cette méthode consiste à modéliser le signal observé par la somme du signal désiré à estimer et la partie réverbérée du signal source. Notre objectif est d'estimer un filtre qui réduit la partie réverbérée dans le signal observé. Le temps de traitement pour cette méthode est considéré élevé pour notre application. Pour cela, on présente une implémentation rapide qui se base sur le calcul de la fonction d'autocorrélation du signal observé. Cette implémentation bénéficie de la transformée de Fourier rapide.

Pour évaluer les deux implémentations, on utilise des indicateurs de performance objectifs pour avoir une idée exacte de leurs performances. On commence par évaluer la méthode de base avec une source de bruit blanc et des signaux de parole supposés stationnaires. Ensuite, on évalue la performance de la méthode de base et la méthode rapide avec des signaux de commande.

Les résultats de l'évaluation de deux méthodes serviront pour étudier les différents facteurs

qui peuvent améliorer le temps de calcul et la qualité du signal à estimer.

ABSTRACT

When a sound source emits an audio signal in a closed room and the microphone is located at a certain distance from the source, the microphone signal is composed of several delayed and attenuated versions of the source signal. In effect, the observed signal is composed of the direct signal coming from the source and its reflections from existing objects in the same room. This phenomenon is called reverberation. Several studies have shown that reverberation causes degradation of the quality of the speech signal and can impair its intelligibility. For this reason, our partner fluent.ai is looking for a solution to the reverberation problem to improve the performance of its automatic speech recognition system. The dereverberation problem is more complicated than the background noise reduction problem because the reverberations we want to reduce are strongly correlated with the desired signal we want to estimate.

It is in this context that we are conducting this research project. Our objective is to study the effect and characteristics of reverberation in order to implement a solution that will reduce it. Since the dereverberation solution will be integrated in an automatic speech recognition system, we are looking for a method that can work in real time and that uses the signal captured by one or more microphones.

We start by studying reverberation, its characteristics and its effects on speech signals. We also present a state of the art of the most known techniques of dereverberation. Among the dereverberation methods presented, the linear prediction dereverberation method will be selected and explained.

We will present the dereverberation method by linear prediction with the model proposed in [1]. This method consists in modelling the observed signal by the sum of the desired signal to estimate and the reverberated part of the source signal. Our goal is to estimate a filter that reduces the reverberated part in the observed signal. The processing time for this method is considered high for our application. For this purpose, we present a fast implementation that is based on the computation of the autocorrelation function of the observed signal. This implementation benefits from the fast Fourier transform.

To evaluate both implementations, objective performance indicators are used to get an accurate idea of their performance. First, we evaluate the basic method with a white noise source and speech signals assumed to be stationary. Then, we evaluate the performance of the basic method and the fast method with control signals.

The results of the evaluation of two methods will be used to study the different factors that can improve the computation time and the quality of the signal to be estimated.

TABLE DES MATIÈRES

LISTE DES TABLEAUX

LISTE DES FIGURES

LISTE DES SIGLES ET ABRÉVIATIONS

ASR	Automatic Speech Recognition
NLP	Natural Language Processing
ASIR	Automatic Speech to Intent Recognition
RI	Réponse Impulsionnelle
RT60	Temps de réverbération
RRD	Rapport de Réverbération Directe
DSB	Delay and Sum Beamformer
MINT	Multiple input/output INverse Theorem
FFT	Fast Fourrier Transform
TFD	Transformée de Fourrier Discrète
SNR	Signal to Noise Ratio
DRR	Direct to Reverberant Ratio
PESQ	Perceptual Evaluation of Speech Quality
MOS	Mean Opinion Score
SNRI	Signal to Noise Ratio Improvement

LISTE DES ANNEXES

CHAPITRE 1 INTRODUCTION

Les travaux sur la reconnaissance automatique de la parole (en anglais, *Automatic Speech Recognition* (ASR)) sont apparus depuis le XX^e siècle. Mais, avant cette période, en 1791, l'inventeur hongrois Wolfgang von Kempelen a conçu un synthétiseur vocal mécanique qui pourrait générer des voyelles, des consonnes et quelques mots comme "papa" et "mama" [4]. Après ce travail, les chercheurs ont commencé à s'intéresser de plus en plus au domaine du traitement de la parole et la première apparition d'un synthétiseur électrique a été en 1939. Ce synthétiseur a été développé par Homer Dudley, ingénieur chez Bell Labs [5]. Par la suite, les applications de traitement de la parole se sont orientées vers la reconnaissance vocale. Le premier système qui a réussi à reconnaître un signal vocal a été fabriqué en 1952. Il s'agit d'un système électronique développé par Davis, Biddulph et Balashek. Ce système permet de reconnaître les chiffres entre zéro et dix avec une précision entre 97 et 99 pourcent [6]. À la fin des années 80, l'apparition du modèle de Markov caché (en anglais, *Hidden Markov Model* (HMM)) a amélioré la performance des systèmes de reconnaissance vocale pour leur permettre de reconnaître des phrases. Parmi les problèmes rencontrés, la difficulté était de trouver une solution générale qui consiste à comprendre différentes personnes qui disent la même chose, mais avec des accents et des dialectes divers. La reconnaissance automatique de la parole est aussi très sensible au bruit généré dans la salle où se trouve la source de la parole et le microphone. Pour qu'un algorithme de traitement de la parole puisse donner de bons résultats, il est indispensable que le signal de parole soit clair et intelligible. Dans la réalité, si on parle dans une salle, notre signal de parole est toujours soumis à des modifications lors de son observation par des microphones. Ces modifications peuvent être dues au bruit de fond aléatoire, comme elles peuvent être générées suite à la réverbération produite par les parois et les objets de la salle. Actuellement, il existe plusieurs solutions efficaces pour réduire le bruit de fond. Par exemple, dans [7], l'auteur propose une méthode de réduction du bruit de fond par soustraction spectrale. La méthode consiste à estimer le spectre du bruit pour le soustraire du spectre du signal observé par le microphone. Pour le problème des réverbérations, plusieurs recherches ont été effectuées sur ce phénomène. Ce problème est plus compliqué que le problème du réduction du bruit de fond car les réverbérations sont très corrélées avec le signal source. C'est dans ce contexte que ce projet de recherche s'effectue en collaboration avec notre partenaire fluent.ai. Il consiste à implémenter un algorithme de déréverbération dans le but d'améliorer la performance de leur système de reconnaissance automatique de la parole. Dans la partie suivante, on présente l'organisme d'accueil de ce projet de recherche, le contexte général de l'application à implémenter, la problématique et

nos objectifs.

1.1 Présentation de l'organisme d'accueil

Depuis sa fondation en 2015, la société fluent.ai se spécialise dans le développement des solutions destinées pour la reconnaissance automatique de la parole. La solution proposée par cette société est basée sur des réseaux de neurones qui peuvent identifier des commandes bien spécifiques telles que "turn on the light" ou "turn off the light". Les systèmes de reconnaissance automatique de la parole typique se composent de deux parties majeures. La première partie consiste à convertir le signal de la parole à un texte. La deuxième partie essaie de traduire le texte en action. C'est ce qu'on appelle le traitement naturel de la langue (en anglais, *Natural Language Processing* (NLP)). La solution proposée par fluent.ai permet de convertir le signal de la parole directement en une action sans avoir besoin de passer par le traitement naturel de la langue. Le traitement nécessaire pour la reconnaissance de la parole se fait sur une carte embarquée sans accès à l'internet. Les produits de fluent.ai sont dédiés à diverses applications comme les interrupteurs d'éclairage ou les cafetières intelligentes.

Le système de reconnaissance automatique de la parole proposé par fluent.ai peut être intégré dans divers dispositifs. Il y a donc des situations où le signal de la parole sera affecté par le bruit de fond et les réverbérations. En effet, plusieurs études ont montré que la réverbération affecte l'intelligibilité de la parole et peut dégrader la performance d'un système de reconnaissance de la parole [8] [9]. Dans ce contexte, fluent.ai cherche à réduire les effets nuisibles de la réverbération sur leur système.

1.2 Contexte général de l'application

La parole est un moyen facile d'interaction entre l'homme et la machine. L'utilisation de commandes vocales peut être très utile dans les situations où on ne peut pas utiliser nos mains (par exemple, le port de gants en usine). L'intérêt pour le développement de la reconnaissance automatique de la parole s'est aussi accru à cause de la prolifération des appareils équipés de microphones. Notre partenaire fluent.ai développe un système de reconnaissance automatique de la parole qui est basé sur le principe *Automatic Speech to Intent Recognition* (ASIR). Ce système peut être intégré dans des appareils dans le but d'effectuer des tâches précises à partir des commandes vocales. Parmi les dispositifs les plus connus pour la reconnaissance automatique de la parole, on cite le système Siri qui est intégré dans les téléphones iPhone, le système Alexa d'Amazon et le Google Home de Google. Les produits de notre partenaire sont

destinés à être intégrés, par exemple, dans des appareils d'une maison intelligente ou dans un système de communication mains libres dans une voiture. Le système de reconnaissance vocale doit interpréter correctement les commandes vocales, idéalement dans plusieurs langues. De plus, ce système doit être très économique en énergie. Cela implique qu'il s'active seulement en cas de besoin et que les calculs sont rapides et efficaces. Contrairement aux systèmes qui font le calcul dans le Cloud par l'accès à internet, le système développé par fluent.ai fait le calcul localement et il ne nécessite pas une connexion à l'internet. Le dispositif ASIR de fluent.ai est équipé de plusieurs microphones (1 à 6). Le nombre et la configuration spatiale des microphones peuvent varier suivant l'application. Ce dispositif est relativement petit et les microphones sont donc assez proches les uns des autres. La situation typique pour passer une commande vocale au système de reconnaissance de la parole est la suivante : une personne peut donner la commande vocale "Start coffee machine", en même temps, le système peut recevoir d'autres signaux vocaux produits par d'autres personnes en arrière-plan. En outre, il peut y avoir aussi les réverbérations générées par les objets existants dans la salle. Ces réverbérations sont des versions retardées et atténuées du signal de parole. Notre partenaire souhaite isoler le signal d'intérêt à partir du signal observé par le microphone dans le but d'améliorer l'intelligibilité de la commande vocale.

1.3 Problématique et objectifs

Comme mentionné précédemment, l'un des problèmes qui affectent l'intelligibilité du signal de la parole est la réverbération. Ce phénomène apparait quand un signal audio est enregistré dans un espace fermé. Si le locuteur est éloigné du microphone, le signal source est généralement modifié par ses réflexions. Les réflexions du signal source causées par les murs, le sol et les objets existants dans la salle persistent après l'interruption de la source de parole. La réverbération provoque une distorsion du signal de parole et elle cause une dégradation considérable de la performance du système de reconnaissance automatique de la parole. Le problème consiste à restaurer le signal source à partir du signal réverbéré capté par le microphone. Ce processus s'appelle la déréverbération. Le deuxième problème qui nous intéresse est que pour la plupart des systèmes de reconnaissance automatique de la parole, le traitement de la parole se fait à distance en utilisant la technologie Cloud par accès internet. Ces systèmes font leur traitement à distance, car la reconnaissance automatique de la parole peut nécessiter beaucoup de puissance de calcul. Notre partenaire cherche à trouver une solution de déréverbération rapide et efficace qui fonctionne sans accés à l'internet. La solution devrait être aussi économique en consommation d'énergie et avoir un temps de latence inférieur à 100 ms pour que l'utilisateur ne remarque pas le retard de traitement.

Pour bien comprendre la déréverbération, on commencera par étudier le phénomène de la réverbération dans le chapitre 2. On présentera aussi ses caractéristiques et son impact sur la parole. Dans le même chapitre, différentes méthodes de déréverbération seront étudiées et analysées pour choisir une méthode appropriée à notre problématique.
Dans le chapitre 3, on présentera la méthode adaptée pour l'implémentation de la solution de déréverbération. On commencera par définir le modèle statistique adopté pour exprimer les signaux considérés. On expliquera ensuite la méthode d'estimation du signal désiré. Enfin, on citera les mesures qui seront utilisées pour évaluer la performance de notre solution.
L'évaluation de la méthode considérée pour la déréverbération sera présentée dans le chapitre 4. C'est dans ce chapitre qu'on présentera les différents tests effectués dans le but d'évaluer l'efficacité de la méthode.

CHAPITRE 2 ÉTAT DE L'ART

2.1 La réverbération

La réverbération est un phénomène connu par l'humanité depuis la préhistoire. Lorsque les gens ont vécu dans des grottes, ils ont été intéressés par l'effet de la réflexion de leurs paroles [10]. Dans *La république*, Platon fait référence à la parole réfléchie par les murs, ce qui implique une compréhension de la réverbération depuis longtemps [8]. Les premières études scientifiques sur la réverbération ont commencé au milieu du XXe siècle, avec des pionniers tels que Bolt [11] et Haas [12]. La réverbération peut être utile dans la vie et peut avoir des avantages dans différentes applications. Parmi ses avantages, on cite le traitement de la musique. Parfois, la réverbération du son peut améliorer l'ambiance générale pour certains types de musique. L'utilité ou la nocivité de la réverbération dépend de l'application [8]. Actuellement, la demande pour des applications qui fonctionnent par la reconnaissance automatique de la parole est en forte augmentation. Le traitement de la réverbération a permis le développement du fonctionnement des dispositifs à commande vocale tels que les appareils mobiles, les ordinateurs portables et les systèmes embarqués dans les voitures. La déréverbération intervient alors pour réduire les réverbérations qui affectent la performance des systèmes de reconnaissance automatique de la parole, surtout pour les applications où le locuteur ou la source acoustique sont éloignés du microphone.

2.1.1 Définition de la réverbération

Lorsqu'une source de parole est située à une certaine distance du microphone dans un espace fermé, le signal capté par le microphone est composé de superpositions retardées et atténuées du signal source. En effet, le signal observé par le microphone est formé du signal direct qui est le signal source et des réflexions générées par les murs, les plafonds ou tout type d'objet qui existe dans la salle. Ce phénomène s'appelle la réverbération. La persistance des réflexions après l'interruption de la source audio peut réduire considérablement les performances des systèmes de reconnaissance automatique de la parole. La figure 2.1 illustre le phénomène de la réverbération.

Il existe deux types de réflexions : En premier lieu, les réflexions qui arrivent juste après le signal direct avec un faible retard. Ces réflexions s'appellent les réflexions précoces [13]. En deuxième lieu, les réverbérations tardives qui ont été générées par plusieurs réflexions du signal direct avant d'atteindre le microphone. Plus la durée d'arrivée des réverbérations

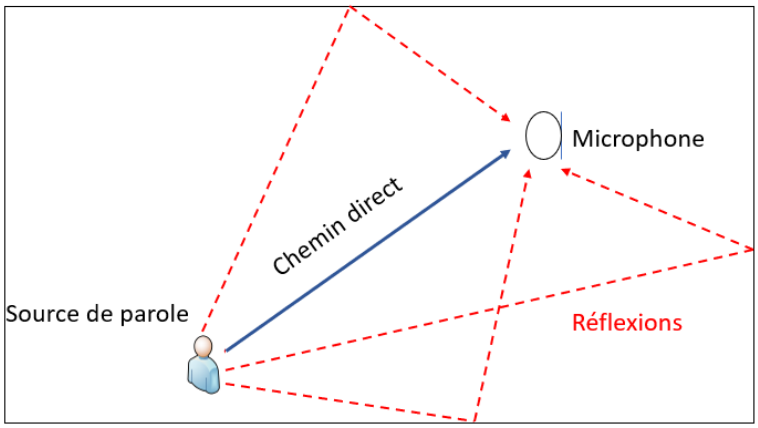

Figure 2.1 Illustration de la réverbération dans un espace fermé.

tardives augmente, plus elles se condensent et se rapprochent les unes des autres au cours du temps. Cette partie des réverbérations affecte les systèmes de reconnaissance de la parole car elle dégrade la qualité du signal observé [3].

2.1.2 Caractéristiques de la réverbération

On note $h(n)$ la réponse impulsionnelle (RI) de la salle. Cette réponse est le signal capté par le microphone si la source est une impulsion de Dirac. Elle dépend des caractéristiques de la salle et de la position du microphone. Cette réponse impulsionnelle est considérée comme un filtre qui s'applique au signal source $s(n)$ pour produire le signal observé $x(n)$ défini par :

$$\begin{aligned} x(n) &= \sum_{k=0}^{N-1} h(k)\ s(n-k), \\ &= \underbrace{h(0)s(n)}_{\text{Signal direct}} + \underbrace{\sum_{k=1}^{D-1} h(k)\ s(n-k)}_{\text{Réflexions précoces}} + \underbrace{\sum_{k=D}^{N-1} h(k)\ s(n-k)}_{\text{Réverbérations tardives}}, \end{aligned}$$

où D est la durée qui sépare les réflexion précoces des réverbérations tardives.
La figure 2.2 montre un exemple de réponse impulsionnelle. La partie qui est en rouge représente les réflexions précoces. Les réflexions dans cette partie sont des pics à peu près séparés les uns des autres. La partie qui est en bleu représente les réverbérations tardives. Plus le temps augmente, plus les réflexions sont proches les unes des autres. Les réverbérations tardives ont une apparence désordonnée et aléatoire.
Chaque réponse impulsionnelle est caractérisée par un paramètre qui s'appelle le temps de

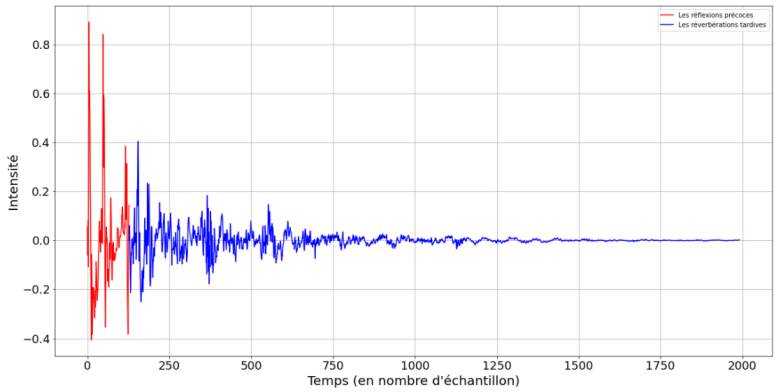

Figure 2.2 Exemple de réponse impulsionnelle.

réverbération, noté RT_{60}. Ce paramètre correspond à la durée nécessaire pour observer une diminution de 60 dB de l'énergie du signal à partir de l'interruption de la source. Le calcul de ce paramètre a été proposé par Sabine [14]. Il est défini par :

$$RT_{60} = \frac{3\ln(10)}{\delta},$$

où δ est une constante qui dépend du type de la salle. Sabine a montré aussi dans ses travaux [15] que le temps de réverbération dépend de la surface de la salle S, de son volume V, du coefficient d'absorption moyen a des parois de la salle et de la vitesse du son c [13]. Il s'obtient avec la formule

$$RT_{60} = \frac{4\ln(10^6)}{c}\frac{V}{Sa}. \tag{2.1}$$

La valeur de RT_{60} varie entre 0,3 s (chambre à coucher) et 10 s (grande église) [14].

2.1.3 Effets de la réverbération sur la parole

On a vu dans la section précédente que la réponse impulsionnelle est composée de trois parties : un premier grand pic qui est le signal direct, une série des réflexions séparées les unes des autres qui correspondent aux réflexions précoces et finalement les réflexions tardives. Les réflexions précoces arrivent juste après le signal direct et elles ne sont pas nombreuses. Après un certain délai (0-50 ms) [8], on observe les réverbérations tardives qui sont très proches les unes des autres et décroissantes au cours du temps. Dans plusieurs articles, les premières réflexions ne sont pas considérées comme des réflexions nuisibles tant qu'elles ne dépassent pas 50 ms. Elles peuvent améliorer l'intelligibilité de la parole si on les intègre au signal direct [16]. Tant que les premières réverbérations sont proches du signal direct, elles permettent de le renforcer et de l'amplifier. C'est pour cela qu'établir une discussion dans un

espace fermé est plus facile qu'en plein air.
Par contre, les réverbérations tardives causent une dégradation de la qualité du signal de parole et son intelligibilité. Elles sont des superpositions atténuées et retardées du signal source. Ces réverbérations restent audibles pendant un certains temps après l'interruption de la source. Parmi les effets des réverbérations tardives sur le signal de la parole, les variations de l'amplitude dans le signal capté par le microphone sont plus lentes que celles du signal source. On peut observer cette modification dans la figure 2.3. La réponse impulsionnelle utilisée ici est celle représentée dans la figure 2.2. En comparant l'onde du signal source et celle du signal réverbéré, on peut voir que les espaces qui existent entre les échantillons du signal source ont été remplis dans le signal réverbéré. C'est comme un chevauchement entre les échantillons du signal réverbéré. Ce chevauchement rend le signal de parole plus difficile à comprendre. L'effet de la réverbération peut s'aggraver si la distance entre le locuteur et le microphone augmente ou si on est dans un milieu très réverbérant.

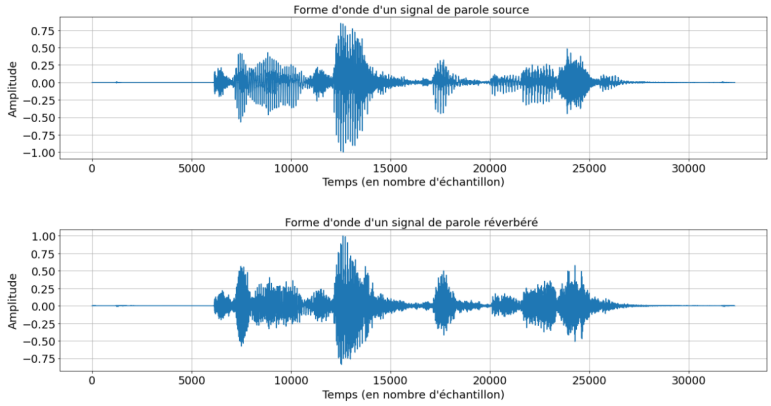

Figure 2.3 Forme d'onde d'un signal de parole (fréquence d'échantillonnage 16 kHz), Signal source en haut et le signal réverbéré en bas.

La réverbération provoque un phénomène qui s'appelle le masquage de superposition. Ce masquage se présente lorsqu'il existe un recouvrement entre deux phonèmes. Pour un signal de parole à rythme constant (sans silence entre les mots), l'énergie de la réverbération de la fin d'un mot vient se superposer à celle du mot suivant. On distingue alors moins bien la différence entre les mots et l'intelligibilité de la parole se dégrade. Cet effet affecte aussi les systèmes de reconnaissance automatique de la parole. Certains de ces systèmes utilisent le spectre et l'enveloppe temporelle des signaux de paroles pour déduire les messages contenus [17]. L'évaluation de leurs performances se fait en calculant un taux d'erreur entre les mots détectés par le système et les mots contenus dans le signal de la parole. Dans [2], Habets montre que le taux d'erreur augmente considérablement si la distance entre le microphone et le locuteur augmente. Il observe aussi que les réverbérations tardives provoquent une aug-

mentation du taux d'erreur pour les systèmes de reconnaissance de la parole.
Les réverbérations tardives sont plus nocives que les réflexions précoces. Dans la partie suivante, on présente quelques méthodes de déréverbération.

2.2 Méthodes de déréverbération

Il existe plusieurs façons de classifier les différents types de méthodes de déréverbération. Par exemple, il y a celles qui sont implémentées dans le domaine temporel et d'autres dans le domaine fréquentiel. Il y a aussi des algorithmes qui fonctionnent avec un seul microphone et d'autres qui nécessitent plus qu'un microphone. Une autre classification peut être citée qui consiste à étudier ce qu'on cherche à estimer. Si notre objectif est d'estimer le signal désiré à partir du signal observé par le microphone, alors on parle de techniques de suppression de la réverbération. Si on veut estimer la réponse impulsionnelle de la salle et par la suite appliquer un filtrage inverse au signal observé, alors on parle de techniques d'annulation de la réverbération [2]. Dans la section suivante, on présente trois familles d'algorithmes dédiés à l'amélioration de la performance des systèmes de reconnaissance de la parole : le premier groupe illustre des méthodes d'amélioration de la parole (en anglais *Speech Enhancement*). La deuxième famille regroupe des méthodes qui consistent à améliorer le spectre de la parole (en anglais *Spectral Enhancement*). La troisième famille inclut des méthodes qui visent spécialement la déréverbération.

2.2.1 Méthodes de rehaussement de la parole

Ces méthodes visent à réduire à la fois le bruit et la réverbération sans avoir besoin d'estimer la réponse impulsionnelle de la salle. Les méthodes de rehaussement de la parole sont moins efficaces que les méthodes dédiées à la déréverbération, mais elles peuvent améliorer le rapport signal sur bruit suffisamment pour être utiles. La technique la plus populaire des méthodes pour le rehaussement de la parole est la technique de formation de faisceaux (en anglais *beamforming*) [18].
La méthode de formation de faisceaux est parmi les premières méthodes de traitement multicanal pour l'amélioration des signaux de parole générés dans des environnements bruyants et réverbérants [19]. Cette méthode vise à maximiser le rapport signal sur bruit et d'améliorer le rapport *Direct to Reverberant Ratio* (DRR). Elle consiste à estimer la direction d'arrivée de l'onde acoustique. Par la suite, en utilisant les acquisitions à partir de M microphones, on essaie de compenser le retard entre les différents temps d'arrivée des différents signaux. Puis, on pondère les signaux et on les additionne pour donner à la fin un signal monocanal. Cette technique s'appelle le *Delay and Sum Beamformer* (DSB). La figure 2.4 illustre cette

technique.

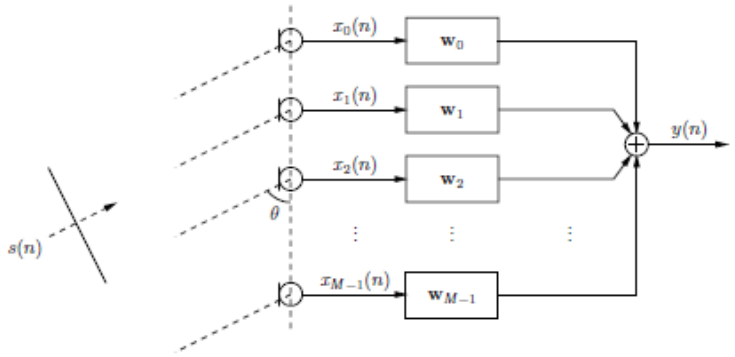

Figure 2.4 La technique Delay and Sum Beamformer [2]

Le signal de sortie de DSB est exprimé comme suit :

$$\hat{x}(n) = \sum_{m=1}^{M} w_m \; x_m(n - \tau_m), \tag{2.2}$$

avec τ_m est le délai de propagation du signal de la source au microphone m et w_m est la pondération appliquée au microphone m. En effet, les composantes venant du trajet direct du signal source sont additionnées, tandis que les composantes dues à la réverbération et au bruit sont atténuées [8]. Les pondérations appliquées aux différents signaux observés par les microphones peuvent être fixes (on privilégie toujours la même direction d'arrivée de l'onde acoustique), dans ce cas on parle de *Beamforming fixe*. Ces pondérations peuvent être aussi adaptées à l'environnement où se trouvent la source et les microphones, on parle de *beamforming adaptatif*. En général, la technique de formation de faisceaux est efficace pour les applications visant à supprimer le bruit additif [20] et sa performance augmente en fonction du nombre de microphones. Cependant, les réverbérations seront partiellement réduites puisqu'elles arrivent de toutes les directions possibles dans une salle. Un autre inconvénient de formation de faisceaux est que l'estimation de la direction d'arrivée du signal direct (en anglais *Direction Of Arrival* (DOA)) est généralement difficile. Une petite erreur au niveau de l'estimation de DOA peut entraîner une détérioration significative de la performance. De plus, la géométrie du réseau de microphones a un grand impact sur la performance [21].

2.2.2 Méthodes par soustraction spectrale

La soustraction spectrale a été largement utilisée pour la déréverbération des signaux de parole. Elle a été proposée par Lebart et al [22]. Dans [22], Lebart et al ont proposé la

soustraction spectrale en utilisant un seul microphone dans le but de réduire le bruit dans le signal de parole. La soustraction spectrale s'effectue en estimant la puissance du bruit ; par la suite, on la soustrait de la puissance du signal réverbéré. Cette méthode a été développée pour être appliquée avec plusieurs microphones [23]. L'algorithme proposé utilise un modèle statistique de la réponse impulsionnelle du canal pour pouvoir calculer sa densité spectrale. Ainsi, l'effet de la réverbération peut être atténué par soustraction spectrale. Le modèle statistique utilisé pour exprimer la réponse impulsionnelle a été proposé dans [24]. Polack a exprimé la réponse impulsionnelle comme un processus aléatoire non stationnaire définit par

$$h(t) = \begin{cases} b(t)e^{-\alpha t}, & t \geq 0, \\ 0, & t < 0, \end{cases} \tag{2.3}$$

où $b(t)$ est un bruit blanc gaussien et α est une constante qui dépend du temps de réverbération RT_{60} de la salle. Cette constante est définie par

$$\alpha = \frac{3\ln(10)}{RT_{60}}. \tag{2.4}$$

Avant d'appliquer la soustraction spectrale, il faut d'abord identifier la partie de la réponse impulsionnelle qu'on veut éliminer. La réponse impulsionnelle h peut être divisée en deux parties et peut s'exprimer comme suit :

$$h(t) = \begin{cases} h_d(t), & 0 \leq t < T, \\ h_r(t), & t \geq T. \end{cases} \tag{2.5}$$

Le temps T est choisi pour que h_d soit composé du signal direct et des premières réverbérations alors que h_r décrit les réverbérations tardives. L'algorithme est résumé dans le diagramme de la figure 2.5.

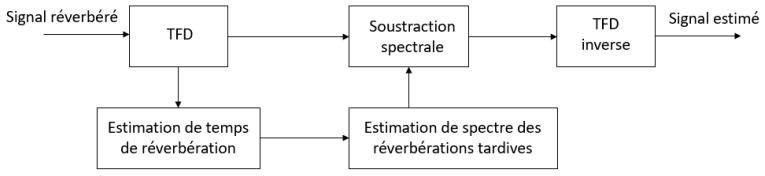

Figure 2.5 La déréverbération par soustraction spectrale

Puisque le paramètre α est calculé à partir du temps de réverbération RT_{60} qui caractérise la salle, le changement de la position de la source n'affecte pas la performance de l'algorithme tant que la source reste dans la même salle. Par contre, l'inconvénient de cette technique

est l'exclusion du spectre de phase qui est important pour la reconnaissance de la parole. La réverbération affecte généralement les spectres de phase et d'intensité du signal source [25]. Pour cette raison, l'efficacité de la déréverbération par la technique de soustraction spectrale est limitée.

2.2.3 Méthodes par déconvolution aveugle

On a vu précédemment que si un signal de parole s est émis dans une salle fermée, alors il sera modifié par la réponse impulsionnelle h de la salle. Le signal observé par le microphone est donné par

$$x(n) = \sum_{k=0}^{N-1} h(k)\ s(n-k).$$

La déréverbération n'est pas un problème de convolution classique où on connaît l'entrée et la sortie. Il s'agit d'un problème où l'entrée est inconnue. Cela veut dire qu'on ne connaît pas le signal de paroles émis, d'où l'appellation de la déréverbération par déconvolution aveugle. L'objectif de la déréverbération par déconvolution aveugle est l'estimation de la réponse impulsionnelle h. Après avoir estimé la réponse impulsionnelle h, on calcule son inverse pour l'appliquer au signal observé afin d'obtenir le signal désiré. La figure 2.6 illustre cette approche.

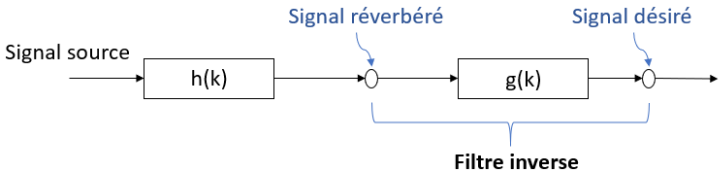

Figure 2.6 Méthode de déréverbération par filtrage inverse

La relation entre le signal δ, la réponse impulsionnelle h et le filtre g est la suivante :

$$\delta(n) = \sum_{k=0}^{N-1} g(k)\ h(n-k),$$

avec

$$\delta(n) = \begin{cases} 1, & \text{si } \ n = 0, \\ 0, & \text{si } \ n \neq 0. \end{cases}$$

L'inconvénient des techniques de déconvolution aveugle est que la réponse impulsionnelle est très sensible aux déplacements du locuteur et aux orientations du microphone. Même le

changement de la température de la salle a un impact sur la réponse impulsionnelle [26]. Parmi les techniques de déconvolution aveugle, on présente la méthode MINT (*Multiple-input/output INverse Theorem*) qui a été introduite par Miyoshi et Kaneda dans [27]. Cette méthode propose l'utilisation de plusieurs canaux pour estimer l'inverse de la réponse impulsionnelle. Pour le cas d'un seul microphone, l'approche du filtre inverse peut ne pas fonctionner, car le filtre peut être instable [28]. Pour le cas de plusieurs microphones, il est possible de calculer l'inverse exact du filtre à condition que les différentes réponses impulsionnelles observées par les microphones ne possèdent aucun zéro en commun. En effet, l'estimé du signal désiré à partir de plusieurs microphones peut s'écrire comme suit :

$$\hat{d}(n) = \sum_{m=1}^{M} (g_m * x_m)(n), \tag{2.6}$$

où M est le nombre de microphones et g_m le filtre à estimer. Pour enlever toute la réverbération et garder juste le signal direct, les g_m doivent satisfaire

$$\delta(n) = \sum_{m=1}^{M} (g_m * h_m)(n), \tag{2.7}$$

où $\delta(n)$ est la fonction impulsion. On peut exprimer l'équation (2.7) dans le domaine de la tranformée en Z comme suit :

$$\sum_{m=1}^{M} G_m(z) H_m(z) = 1. \tag{2.8}$$

L'existence d'une solution de (2.8) est assurée par l'identité de Bézout si les $H_m(z)$ n'ont aucun zéro en commun. Les filtres G_m sont calculés par l'algorithme proposé dans [27]. La performance de la technique d'inversion du filtre reste toujours limitée. Cette technique est sensible à la position du locuteur et du microphone. À chaque fois que le locuteur change de position, on est obligé de recalculer la réponse impulsionnelle, ce qui prendra du temps si le locuteur se déplace en permanence.

La méthode de déréverbération qui a attiré notre attention est la méthode de prédiction linéaire. Cette méthode a été citée dans plusieurs références dans lesquelles les auteurs montrent qu'elle donne de bons résultats de déréverbération et qu'elle peut fonctionner avec un ou plusieurs microphones [29] [30]. Cette méthode utilise l'estimation linéaire par maximum de vraisemblance pour estimer la réverbération tardive $r(t)$ à l'instant t à partir du signal microphone $x(t)$ observé dans le passé. Il a été démontré que le résidu $x(t) - r(t)$ est une bonne estimation du signal source. Cette approche semble fonctionner parce que la partie réverbérée du signal à un moment donné est fortement corrélée avec les valeurs du signal observé à d'autres moments. Cette méthode linéaire possède plusieurs avantages pour

notre problème : elle produit une bonne approximation d'une déconvolution aveugle. Elle exploite à la fois les informations d'amplitude et de phase, ce qui améliore la précision et peut être utile pour la reconnaissance automatique de la parole. Elle peut aussi utiliser plusieurs microphones et prendre en considération les différences acoustiques entre les canaux. L'algorithme est récursif et peut s'adapter à des conditions variables telles que le changement de position du locuteur. Elle peut aussi être combinée avec la méthode de formation faisceaux pour effectuer à la fois déréverbération et réduction du bruit. Enfin, il est prouvé que cette méthode de filtrage linéaire peut améliorer les performances du système de reconnaissance automatique de la parole. On présente le principe de cette méthode dans le chapitre suivant.

CHAPITRE 3 DÉRÉVERBÉRATION PAR PRÉDICTION LINÉAIRE AVEC NORMALISATION PAR LA VARIANCE

3.1 Introduction

La méthode de déréverbération par prédiction linéaire avec normalisation par la variance est basée sur la définition d'un modèle statistique du signal source. L'objectif de cette méthode est de réduire la réverbération tardive à partir du signal de parole capté le microphone. Si un signal de parole est enregistré dans un environnement fermé et si la source n'est pas très proche du microphone, le signal source est généralement modifié par des échos générés par les murs, les plafonds et les objets existants dans la salle. En effet, il y a des réverbérations qui atteignent le microphone juste après le signal direct. On les appelle les premières réverbérations. Après l'interruption de la source, il y a aussi des échos qui persistent. Ces échos sont les réverbérations tardives. la figure 3.1 montre un exemple de la réponse impusionnelle.

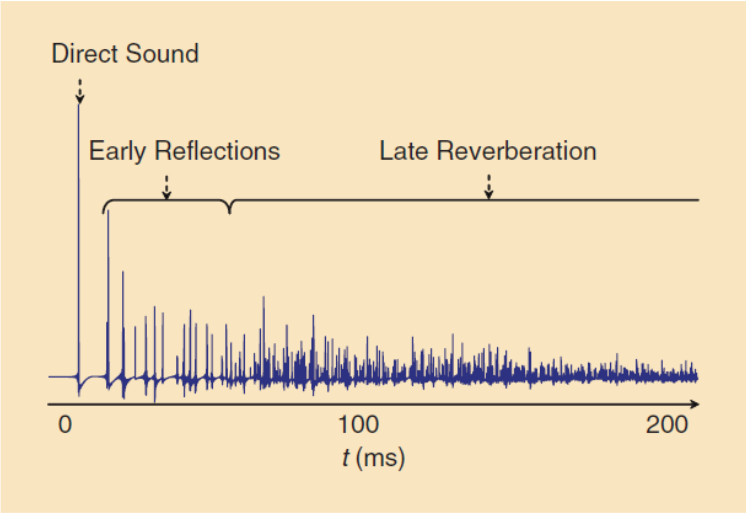

Figure 3.1 Réponse impulsionnelle [3]

La déréverbération par prédiction linéaire permet d'estimer le signal désiré sans connaissance préalable de la réponse impulsionnelle (RI) de la salle. En partant du modèle statistique du signal source, on peut définir le problème de déréverbération en mettant comme objectif la prédiction d'un signal désiré qui a à peu près les mêmes caractéristiques que le signal pur. L'estimation du signal désiré et les paramètres du modèle statistique se font en utilisant une fonction de vraisemblance. Le processus de déréverbération par une approche statistique

passe par ces trois étapes :

1. Définir un modèle statistique pour le signal de la parole
2. Définir une fonction objectif pour déterminer les paramètres du modèle
3. Estimer le signal désiré à partir du signal capté par le microphone

Dans [1], Kinoshita *et al.* optent pour cette approche statistique en supposant que le signal désiré peut être exprimé comme un ensemble de variables gaussiennes indépendantes et identiquement distribuées de moyenne nulle et d'écart type constant. Par la suite, ils utilisent la maximisation de la vraisemblance pour estimer les paramètres du modèle. Pour l'estimation du signal désiré, ils appliquent la méthode de la prédiction linéaire. Deux implémentations ont été proposées dans leur article, la première est dans le domaine temporel et la deuxième est dans le domaine fréquentiel. On s'intéresse à l'implémentation dans le domaine temporel dans ce chapitre.

3.2 Définition du modèle statistique des signaux considérés

On suppose dans cette partie qu'on a un seul signal source qui est un signal de parole et M microphones qui sont un peu distants du locuteur et que le signal de bruit est nul. Le signal source à l'instant n est noté $s(n) \in \mathbb{R}$ avec $n \in T \subset \mathbb{Z}$. On suppose dans la suite que $\mathbb{E}(s(n)) = 0$ pour tout n. Cette hypothèse est justifiée par la nature oscillante des signaux de parole. Le signal capté par le microphone m est appelé le signal observé et il est noté $x_m(n) \in \mathbb{R}$ avec $m \in \{1, 2, ..., M\}$. La réponse impulsionnelle entre la source et le microphone m est de longueur L_h et elle s'exprime comme suit :

$$h_m := [h_m(0), h_m(1), ..., h_m(L_h - 1)]^T \in \mathbb{R}^{L_h}$$

Le signal observé par le microphone m est la convolution de la RI h_m avec le signal source s :

$$x_m(n) = \sum_{k=0}^{Lh-1} h_m(k)\ s(n-k). \tag{3.1}$$

Le signal $x_m(n)$ contient le signal direct de la parole et des échos avec des délais et des intensités variables. Ces échos sont des réflexions directes et indirectes du signal de la parole avec les objets existants dans la salle. On appelle ce phénomène la réverbération. L'objectif de la déréverbération est d'estimer un signal qui contient moins de réverbération que le signal capté par le microphone et qui ressemble au signal source. Dans la suite du travail, on fixe le nombre de microphones à $M = 1$, et le signal de l'unique microphone sera noté $x(n)$.
Le signal observé $x(n)$ peut être exprimé par la somme de trois parties : le signal direct,

le signal des premières réverbérations et le signal des réverbérations tardives. On définit le signal direct et le signal des premières réverbérations comme étant la partie désirée qu'on veut estimer. Elle est notée $d(n)$. Le signal $r(n)$ des réverbérations tardives est la partie qu'on essaie de réduire. Le signal capté par le microphone peut s'exprimer de la manière suivante :

$$x(n) = d(n) + r(n), \tag{3.2}$$

avec

$$d(n) = \sum_{k=0}^{D-1} h(k)\ s(n-k) \tag{3.3}$$

et

$$r(n) = \sum_{k=D}^{L_h-1} h(k)\ s(n-k).$$

Le paramètre $D \in \mathbb{N}$. Par exemple, dans la figure 3.1, on a $D \approx 50ms$. C'est la durée qui sépare les premières réverbérations des réverbérations tardives.

3.3 Méthode d'estimation du signal désiré

On définit

$$X(n) := [x(n), x(n-1), ..., x(n-L_c+1)]^T \in \mathbb{R}^{L_c},$$

$$C := [c(0), c(1), ..., c(L_c-1)]^T \in \mathbb{R}^{L_c}.$$

En utilisant ces définitions, on peut noter le signal observé $x(n)$ comme suit :

$$x(n) = C^T X(n-D) + d(n). \tag{3.4}$$

Le signal désiré est la prédiction résiduelle entre le signal $x(n)$ et la convolution entre le filtre C et une séquence du signal observé dans le passé noté $X(n-D)$. L'estimé $\hat{d}(n)$ du signal désiré $d(n)$ est

$$\hat{d}(n) = x(n) - \hat{C}^T X(n-D), \tag{3.5}$$

où $\hat{C}$ est le filtre à calculer à partir du signal observé $x(n)$. Le principe de cette méthode est basé sur l'estimation des coefficients du filtre C par prédiction linéaire, en utilisant le signal réverbéré afin d'en extraire le signal désiré. La partie suivante représente la méthode d'estimation du modèle statistique utilisée pour exprimer le signal désiré. Par la suite, on définit la fonction de vraisemblance pour estimer les paramètres du modèle et enfin déterminer l'expression de l'estimé du filtre C.

On définit $\sigma(n) := \sqrt{\mathbb{E}[(d(n))^2]}$ et $\Sigma := [\sigma(0), \sigma(1), ..., \sigma(N-1)]^T$. Les paramètres à estimer sont $\theta := (C, \Sigma)$. La fonction de vraisemblance est définie comme suit :

$$L(C, \Sigma) = \prod_{n=0}^{N-1} \frac{1}{\sqrt{2\pi}\ \sigma(n)}\ \exp\left[-\frac{1}{2(\sigma(n))^2}\left(x(n) - C^T\ X(n-D)\right)^2\right]. \tag{3.6}$$

Cette expression suppose que le signal désiré $d(n) = x(n) - C^T X(n-D)$ est une suite de variables aléatoires gaussiennes indépendantes et identiquement distribuées de moyenne nulle et d'écart type $\sigma(n)$. L'écart type $\sigma(n)$ dans (3.6) est estimé dans une fenêtre de largeur L_f centrée à l'instant n. L_f est choisi de telle sorte que le signal de parole soit approximativement stationnaire pendant une durée L_f. Le calcul de l'écart type se fait comme suit :

$$(\sigma(n))^2 := \frac{1}{L_f} \sum_{k=n-L_f/2+1}^{n+L_f/2} (d(k))^2\,. \tag{3.7}$$

Dans [1], les auteurs supposent que le signal source $s(n)$ est gaussien sur des intervalles de dizaines de millisecondes. Ils font l'hypothèse que $s(n)$ est une série de variables gaussiennes avec une moyenne nulle et un écart type presque constant sur des intervalles de longueur L_f. Puisque une somme de variables gaussiennes indépendantes est une variable gaussienne, en regardant l'équation (3.3), on peut dire alors que $d(n)$ est aussi gaussienne. On peut estimer alors le signal désiré $\hat{d}(n)$ par la fonction (3.6). Puisque le signal désiré est considéré gaussien sur des intervalles de longueur L_f, sa variance est constante sur ces intervalles et elle se calcule par la moyenne des valeurs de $d(n)$ comme la montre l'équation (3.7).
Si on applique le logarithme à la fonction de vraisemblance, cette dernière s'écrit

$$\ln(L(\theta)) = \sum_{n=0}^{N-1} -\frac{1}{2(\sigma(n))^2}\left(x(n) - C^T\ X(n-D)\right)^2 - \sum_{n=0}^{N-1} \ln(\sigma(n)) + K,$$

avec $K = -N\ln(2\pi)/2$ qui est une constante. Cette constante n'a pas d'effet sur les conditions d'optimalité, on peut l'omettre dans la suite. Le problème de déréverbération sera résolu si on trouve θ qui maximise la fonction de vraisemblance qui s'écrit comme suit :

$$\ln(L(\theta)) = -\frac{1}{2}\sum_{n=0}^{N-1} \frac{1}{(\sigma(n))^2}\left(x(n) - C^T\ X(n-D)\right)^2 - \frac{1}{2}\sum_{n=0}^{N-1} \ln((\sigma(n))^2). \tag{3.8}$$

Cette équation correspond à l'équation (14) dans [1]. Pour simplifier la notation dans la suite, on définit la variance à l'instant n par $v(n) := \sigma(n)^2$. Le vecteur de la variance est $V := [v(0), v(1), ..., v(N-1)]^T$ et on redéfinit $\theta := (C, V)$. On peut alors écrire l'équation

(3.8) sous la forme

$$\mathcal{L}(\theta) := -\frac{1}{2}\sum_{n=0}^{N-1}\frac{1}{v(n)}\Big(x(n) - C^T\ X(n-D)\Big)^2 - \frac{1}{2}\sum_{n=0}^{N-1}\ln(v(n)). \tag{3.9}$$

Notre objectif est de maximiser $\mathcal{L}(\theta)$, où minimiser -$\mathcal{L}(\theta)$. Les conditions d'optimalité du premier ordre pour ce problème sont :

$$\begin{aligned} \nabla_V \mathcal{L} &= 0, \\ \nabla_C \mathcal{L} &= 0. \end{aligned} \tag{3.10}$$

La première condition dans (3.10) qui représente $\partial\mathcal{L}/\partial v(n) = 0$ pour tout n, donne le résultat suivant :

$$v(n) = \Big(x(n) - C^T\ X(n-D)\Big)^2, \quad n = 0, 1, ..., N-1. \tag{3.11}$$

Ces valeurs de $v(n)$ maximisent la fonction de vraisemblance pour toutes les valeurs de C. En développant la première partie dans l'équation (3.9), $\mathcal{L}(\theta)$ devient

$$-\frac{1}{2}\sum_{n=0}^{N-1}\frac{1}{v(n)}\Big([x(n)]^2 + C^T\ X(n-D)X^T(n-D)\ C - 2\ x(n)\ C^T\ X(n-D)\Big).$$

La deuxième condition de (3.10) donne l'équation suivante :

$$\left[\sum_{n=0}^{N-1}\frac{X(n-D)X^T(n-D)}{v(n)}\right] C = \sum_{n=0}^{N-1}\frac{x(n)}{v(n)}\ X(n-D). \tag{3.12}$$

Les deux équations (3.11) et (3.12) forment un système d'équations non linéaires pour les variables C et V. Si on définit la matrice A par

$$A := \sum_{n=0}^{N-1}\frac{X(n-D)X^T(n-D)}{v(n)} \in \mathbb{R}^{L_c \times L_c}$$

et le vecteur B par

$$B := \sum_{n=0}^{N-1}\frac{x(n)}{v(n)}\ X(n-D) \in \mathbb{R}^{L_c},$$

alors l'équation (3.12) prend la forme :

$$A\ C = B,$$

qui est linéaire en C si les variances V sont connues. Si la matrice A n'est pas inversible,

alors on peut chercher le vecteur $C \in \mathbb{R}$ qui minimise $||A\ C - B||^2$ en utilisant la méthode des moindres carrés. L'estimé de C est alors

$$\hat{C} = (A^T A)^{-1} A^T\ B.$$

Initialement, la valeur de la variance $v(n)$ est calculée à partir du signal observé $x(n)$. Cette valeur nous permet de calculer le premier estimé du signal désiré. Pour la prochaine itération, le calcul de la variance se fait en utilisant le signal désiré estimé à l'itération précédente. L'algorithme de déréverbération est représenté par les étapes suivantes :

1. Initialiser
$$\hat{v}(n) = \max\left\{ \frac{1}{L_f} \sum_{k=n-L_f/2+1}^{n+L_f/2} (x(k))^2, \epsilon \right\} \tag{3.13}$$
où $\epsilon > 0$ est une constante.

2. Répéter (a), (b) et (c) jusqu'à la convergence

(a) Mettre à jour C :

$$A := \sum_{n=0}^{N-1} \frac{X(n-D)X^T(n-D)}{\hat{\sigma}(n)^2} \in \mathbb{R}^{L_c \times L_c}, \tag{3.14}$$

$$B := \sum_{n=0}^{N-1} \frac{x(n)}{\hat{\sigma}(n)^2} X(n-D) \in \mathbb{R}^{L_c}, \tag{3.15}$$

$$\hat{C} = A^{-1}B. \tag{3.16}$$

(b) Mettre à jour $\hat{d}(n)$:

$$\hat{d}(n) = x(n) - \hat{C}^T X(n-D). \tag{3.17}$$

(c) Mettre à jour $\hat{v}(n)$

$$\hat{v}(n) = \max\left\{ \frac{1}{L_f} \sum_{k=n-L_f/2+1}^{n+L_f/2} \left(\hat{d}(k)\right)^2, \epsilon \right\}. \tag{3.18}$$

ϵ prend une valeur très petite pour éviter la division par zéro dans l'algorithme. On a fixé $\epsilon = 10^{-6}$ pour notre cas. Pour l'étape (2), le critère de convergence est lié aux valeurs du filtre C. Si on ne constate plus une évolution au niveau des valeurs de C, on peut dire que l'algorithme a convergé.

3.4 Pré-blanchiment du signal observé

Le signal de parole est caractérisé par une certaine corrélation entre les échantillons sur des petits intervalles (Figure 3.3). Pour rendre le signal de parole observé plus proche d'une séquence des variables indépendantes et identiquement distribuées, on a besoin d'appliquer le pré-blanchiment avant de passer à la déréverbération. Le principe du pré-blanchiment d'un signal consiste à modifier la fonction d'autocorrélation de ce signal pour la rendre similaire à celle d'un signal bruit blanc. Un signal bruit blanc ne possède aucune corrélation entre un échantillon du signal à l'instant n et un échantillon à l'instant $n+L$ (Figure 3.2). La fonction d'autocorrélation est estimée avec

$$\hat{R}_x(n) := \frac{1}{(N-n)\sigma^2} \sum_{m=0}^{N-n-1} x(m)\ x(m+n), \tag{3.19}$$

où σ, l'écart type de $x(n)$, est supposé constant pour $n = 0, 1, ..., N-1$.

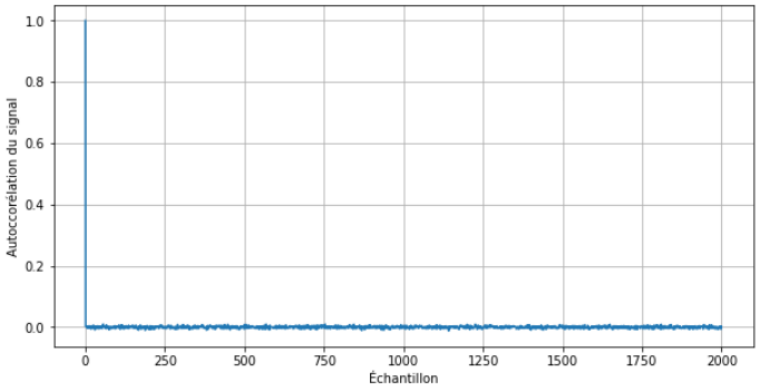

Figure 3.2 Autocorrélation d'un bruit blanc gaussien

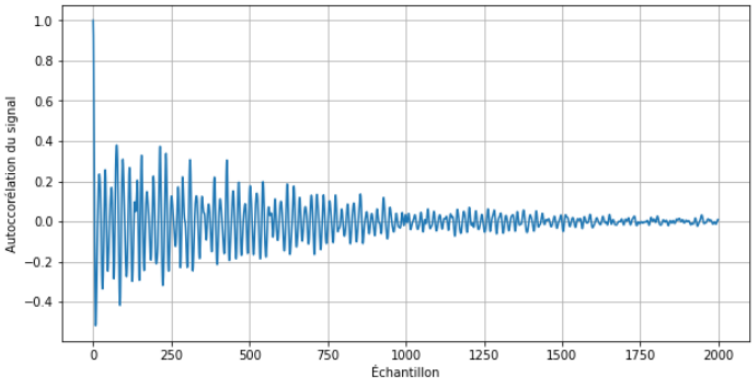

Figure 3.3 Autocorrélation d'un signal de parole

L'opération de pré-blanchiment consiste à décorréler les composantes du signal observé $x(n)$ et d'obtenir des variances unitaires. En d'autres termes, la matrice de covariance du signal obtenu y devrait être égale à la matrice identité :

$$\mathbb{E}(yy^T) = I. \tag{3.20}$$

La méthode utilisée pour appliquer le pré-blanchiment sur le signal observé consiste à calculer une matrice W qui fait en sorte que le signal $y(n) = Wx(n)$ soit un bruit blanc. Cette matrice doit vérifier :

$$\begin{aligned}\mathbb{E}(yy^T) &= \mathbb{E}(Wxx^TW) \\ &= W\mathbb{E}(xx^T)W \\ &= I.\end{aligned}$$

On note $C := \mathbb{E}(xx^T)$. La matrice W se calcule en utilisant la décomposition en valeurs propres de la matrice de covariance $\mathbb{E}(xx^T)$. On définit E comme la matrice orthogonale des vecteurs propres de C et $D = diag(\lambda_1, ..., \lambda_n)$ la matrice diagonale de ses valeurs propres [31]. Par définition, C, E et D sont liés par :

$$CE = ED,$$

d'où

$$E^TCE = E^TED.$$

La matrice E est orthogonale car C est symétrique, donc $EE^T = I$, d'où

$$\begin{aligned} E^T C E &= D, \\ \Rightarrow D^{-1/2}(E^T C E)D^{-1/2} &= D^{-1/2} D D^{-1/2}, \\ \Rightarrow (D^{-1/2}E^T)C(E D^{-1/2}) &= I, \\ (D^{-1/2}E^T)C(D^{-1/2}E^T)^T &= I, \end{aligned}$$

et donc

$$W = D^{-1/2}E^T, \tag{3.21}$$

où $D^{-1/2}$:=diag$(\lambda_1^{-1/2}, ..., \lambda_n^{-1/2})$.

3.5 Implémentation rapide de la méthode

Notre objectif est de développer une solution de déréverbération qui sera rapide et efficace. Suivant [2], on suppose que le processus stochastique $x(n)$ est stationnaire sur des fenêtres de taille entre 20 ms et 30 ms. Plus précisément, on fait l'hypothèse que la fonction d'auto-corrélation $R_x(l)$, qui est définie par

$$R_x(l) := \frac{1}{\sigma^2}\mathbb{E}[x(n)x(n+l)],$$

est indépendante de n pour chaque $l \in 0, 1, ..., L_c - 1$ et $\sigma^2 = Var(x(n))$ est constante pour tout n dans la fenêtre. En particulier, pour $l = 0$ la variance $\mathbb{E}(x(n)^2)$ est aussi constante. Alors, on suppose que la variance est constante pour chaque fenêtre I_k de taille L avec $L \in \mathbb{Z}$. Le support du signal $x(n)$ est divisé en plusieurs intervalles I_k de taille constante L et de variance constante sur chaque intervalle. On définit I comme l'ensemble des intervalles I_k

$$I = \bigcup_{k=1}^{K} I_k,$$

avec $v(n) = v_k$ pour chaque $n \in I_k$. On estime v_k avec

$$\hat{v}_k = \max\left\{\frac{1}{|I_k|}\sum_{n\in I_k}\left(\hat{d}(n)\right)^2, \epsilon\right\}.$$

Puisqu'on peut supposer que le signal de parole est stationnaire par intervalles, on peut alors appliquer la transformée de Fourier rapide (en anglais, *Fast Fourier Transform (FFT)*). La

matrice A est définie par

$$A := \frac{1}{|I|} \sum_{n \in I} \frac{1}{v(n)} X(n-D)X^T(n-D) \in \mathbb{R}^{L_c \times L_c}. \tag{3.22}$$

Si on suppose que la variance est constante sur chaque intervalle I_k, alors A prend la forme

$$A = \frac{|I_k|}{|I|} \sum_{k=1}^{K} \frac{1}{v_k} \frac{1}{|I_k|} \sum_{n \in I_k} X(n-D)X^T(n-D). \tag{3.23}$$

L'élément $A(i,j)$ de la matrice A est

$$A(i,j) = \frac{|I_k|}{|I|} \sum_{k=1}^{K} \frac{1}{v_k} \frac{1}{|I_k|} \sum_{n \in I_k} x(n-D-i)x(n-D-j). \tag{3.24}$$

On définit

$$R_k(l) := \mathbb{E}[x(n)x(n+l)|n \in I_k]. \tag{3.25}$$

Donc, on peut noter

$$\mathbb{E}[x(n-D-i)x(n-D-j)|n \in I_k] := R_k(i-j), \tag{3.26}$$

puisque $R_k(l) = R_k(-l)$ pour une série x qui est stationnaire. Alors, l'espérance de l'élément $A(i,j)$ de la matrice A peut être exprimé par la fonction d'autocorrélation du signal $x(n)$:

$$\mathbb{E}[A(i,j)] = \frac{|I_k|}{|I|} \sum_{k=1}^{K} \frac{1}{v_k} R_k(i-j). \tag{3.27}$$

L'estimé $\hat{A}(i,j)$ de $\mathbb{E}[A(i,j)]$ est

$$\hat{A}(i,j) = \frac{|I_k|}{|I|} \sum_{k=1}^{K} \frac{1}{v_k} \hat{R}_k(i-j), \tag{3.28}$$

où l'estimé $\hat{R}_k(l)$ de $R_k(l)$ est

$$\hat{R}_k(l) := \frac{1}{|I_k|} \sum_{n \in I_k} x(n)x(n+l) = \frac{1}{|I_k|} F^{-1}\left(|\hat{x}(k)|^2\right)(l), \tag{3.29}$$

pour $l = 0, 1, ..., L-1$. On définit

$$\hat{x}(k) := \sum_{n=0}^{L-1} x(n) e^{-i2\pi n \frac{k}{L}} \tag{3.30}$$

la transformée de Fourier discrète (TFD) de x, et F^{-1} dénote la TFD inverse. L'identité (3.29) est une version discrète du théorème de Wiener-Khinchin.
Les étapes du calcul rapide de la matrice A sont les suivantes : on commence par calculer l'estimé $\hat{R}_k(l)$ pour chaque intervalle I_k en utilisant la TFD représentée à l'équation 3.29. Ce calcul bénéficie de la transformée de Fourrier rapide. Ensuite, on calcule la fonction d'autocorrélation normalisée par la variance v_k pour chaque I_k avec

$$\hat{R}(l) := \sum_{k=1}^{K} \frac{1}{v_k} \hat{R}_k(l). \tag{3.31}$$

Enfin, on calcule l'estimé $\hat{A}$ de la matrice A par

$$\hat{A}(i,j) = \frac{|I_k|}{|I|} \hat{R}(i-j). \tag{3.32}$$

La taille L des intervalles I_k doit être constante et choisie de sorte que la variance soit approximativement constante. En effet, on doit prendre $L \geq L_c + D$ car la matrice A est de dimension $(L_c \times L_c)$ et on ne peut commençer le calcul que seulement si on a déja D observations du signal x. Pour le calcul du vecteur B, on applique le même principe que la matrice A. Le vecteur B est défini par

$$B := \frac{1}{|I|} \sum_{n \in I} \frac{x(n)}{v(n)} X(n-D) \in \mathbb{R}^{L_c}. \tag{3.33}$$

En supposant que la variance $v(n)$ est constante par intervalle, le vecteur B s'exprime alors comme suit :

$$B := \frac{|I_k|}{|I|} \sum_{k=1}^{K} \frac{1}{v_k} \frac{1}{|I_k|} \sum_{n \in I_k} x(n) X(n-D). \tag{3.34}$$

Alors, $\mathbb{E}(B)$ est

$$\begin{aligned}
\mathbb{E}(B) &= \frac{|I_k|}{|I|} \sum_{k=1}^{K} \frac{1}{v_k} \frac{1}{I_k} \sum_{n \in I_k} \mathbb{E}\left[x(n)(x(n-D), x(n-D-1), ..., x(n-D-(L_c-1)))^T \right], \\
&= \frac{|I_k|}{|I|} \sum_{k=1}^{K} \frac{1}{v_k} \frac{1}{I_k} \sum_{n \in I_k} \mathbb{E}\left[x(n)x(n-D), x(n)x(n-D-1), ..., x(n)x(n-D-(L_c-1))^T \right], \\
&= \frac{|I_k|}{|I|} \sum_{k=1}^{K} \frac{1}{v_k} (\hat{R}_k(D), \hat{R}_k(D+1), ..., \hat{R}_k(D+L_c-1))^T, \\
&= \frac{|I_k|}{|I|} (\hat{R}(D), \hat{R}(D+1), ..., \hat{R}(D+L_c-1))^T.
\end{aligned}$$

L'élément $B(i)$ du vecteur B est donné par

$$\hat{B}(i) = \frac{|I_k|}{|I|}\hat{R}(D+i),\ i \in \{0, 1, ..., L_c - 1\}\,. \tag{3.35}$$

3.6 Mesures de déréverbération

On a besoin de trouver une mesure pour évaluer la performance de l'algorithme de la déréverbération. C'est vrai qu'en écoutant le signal réverbéré et le signal déréverbéré on peut commenter l'efficacité de la déréverbération, mais cela n'est pas toujours facile ou même possible, et ça n'est pas toujours objectif. Pour cela, on a besoin d'utiliser des mesures quantitatives qui nous donnent une idée claire sur l'efficacité de l'algorithme. Dans cette section, on présente les mesures les plus utilisées pour évaluer la qualité d'une méthode de réduction de bruit.

3.6.1 Rapport Signal sur Bruit

Si s est le signal et w est le bruit :
Le rapport signal-bruit initial est noté $\text{SNR}_{initial} = \frac{\|s\|_2}{\|w\|_2}$. Dans la suite, nous utiliserons l'abréviation SNR pour "Signal To Noise Ratio". Le rapport signal-bruit final, c'est-à-dire après le traitement, est noté $\text{SNR}_{final} = \frac{\|s\|_2}{\|\hat{s}-s\|_2}$. Alors, le SNR 'improvement' est noté SNRI := $\frac{\text{SNR}_{final}}{\text{SNR}_{initial}} = \frac{\|w\|_2}{\|\hat{s}-s\|_2}$. $\hat{s}$ est le signal estimé après déréverbération et le symbole $\|.\|_2$ représente la norme euclidienne. Cette mesure nous permet de savoir la quantité de bruit dans le signal estimé.

3.6.2 Rapport de Réverbération Directe

Le rapport de réverbération directe nécessite la connaissance de la réponse impulsionnelle estimée à partir du signal déréverbéré. Alors, on n'a pas besoin de connaitre le signal source. Cette mesure est définie par :

$$\text{RRD} = \frac{\sum_{n=0}^{n_D} h^2(n)}{\sum_{n=n_D+1}^{\infty} h^2(n)}, \tag{3.36}$$

$h(n)$ représente la réponse impulsionnelle estimée à partir du signal déréverbéré et n_d représente l'instant d'arrivée du signal direct et les premières réverbérations au microphone [13].

3.6.3 Perceptual Evaluation of Speech Quality (PESQ)

Cette mesure est très utilisée dans le domaine des télécommunications. Elle permet d'évaluer la qualité de la parole et elle a été standardisée par la recommandation de l'UIT-T P.862 [32], qui est l'Union Internationale des Télécommunications et qui assure plusieurs recommandations utilisées dans ce domaine. L'objectif de cette mesure est de comparer la qualité du signal observé par rapport à celle du signal source. La valeur de PESQ est très proche de la valeur de la note d'opinion moyenne (en anglais, *Mean Opinion Score* ou MOS) qui est une mesure subjective sur la qualité de la parole. Théoriquement, le score PESQ est une combinaison linéaire entre la perturbation moyenne d_{sym} et la perturbation asymétrique moyenne d_{asym} [9] :

$$\text{PESQ} = a_0 + a_1 d_{sym} + a_2 d_{asym}, \tag{3.37}$$

où a_0, a_1 et a_2 sont des coefficients à estimer par des régressions linéraires avec ces trois facteurs : La distorsion de la parole, la distorsion du bruit et la qualité globale de la parole. Le score PESQ va de 1 à 4.5. La valeur 1 signifie que la qualité de la parole est mauvaise, 4.5 signifie qu'elle est excellente.

3.6.4 Mesure d'amélioration de la qualité du signal désiré

On note $\hat{r}(n)$ l'estimé de $r(n)$ qui est défini par :

$$\hat{r}(n) := C^T X(n - D).$$

L'estimé $\hat{d}(n)$ du signal désiré $d(n)$ est :

$$\hat{d}(n) = x - \hat{r} = d + r - \hat{r}, \tag{3.38}$$

cela veut dire qu'on soustrait l'estimé de la réverbération tardive du signal désiré pour réduire la réverbération. On obtient une réduction de la réverbération si

$$||r - \hat{r}||^2 < ||r||^2 \Longleftrightarrow \frac{||r - \hat{r}||^2}{||r||^2} < 1.$$

La qualité d'un signal est améliorée si le rapport signal sur bruit (SNR) augmente. Le signal qu'on veut améliorer est le signal désiré. Selon l'équation (3.2), le rapport initial du signal sur bruit est $\text{SNR}_i = \frac{||d||^2}{||r||^2}$. En appliquant la déréverbération, le rapport signal sur bruit final

est $\mathrm{SNR}_f = \frac{||d||^2}{||r-\hat{r}||^2}$. L'amélioration de la qualité du signal est alors calculée comme suit :

$$\mathrm{SNRI} := \frac{\mathrm{SNR}_f}{\mathrm{SNR}_i} = \frac{||r||^2}{||r-\hat{r}||^2}.$$

On peut dire qu'il y a une amélioration au niveau de la qualité du signal si et seulement si $\mathrm{SNRI} > 1$. Il est préférable d'avoir un SNR_f aussi grand que possible, et de préférence supérieur à 1.

CHAPITRE 4 RÉSULTATS EXPÉRIMENTAUX

Dans ce chapitre, on va présenter les différents tests effectués. Pour évaluer la performance de l'algorithme choisi, on utilisera les mesures présentées dans le chapitre 2. La plupart des mesures objectives nécessitent la connaissance à priori du signal source et de la réponse impulsionnelle de la salle. À cette fin, notre partenaire fluent.ai a créé une base de données composée de 8 signaux de paroles enregistrés dans une salle anéchoïque. Ces signaux sont des commandes vocales que fluent.ai considère pour tester leur système de reconnaissance de la parole. La durée de ces signaux varie entre 1 et 4 secondes et leur fréquence d'échantillonnage est de 16 kHz. Toutes ces énonciations sont en anglais. Pour produire l'effet de la réverbération, on convolue le signal de parole avec une réponse impulsionnelle qui modélise la réverbération. Pour cette raison, fluent.ai a créé une autre base de données qui contient des réponses impulsionnelles (RI) enregistrées dans différents environnements. Les RI sont classées en deux catégories. La première contient des RI qui ont été enregistrées dans un couloir, une salle de bain ou dans un amphithéâtre, où la réverbération est forte. La seconde contient des RI qui ont été enregistrées dans des petites salles en absence d'éléments réfléchissants, où la réverbération est faible.

4.1 Performance de l'estimateur pour une source de bruit blanc

On considère un signal source qui est un bruit blanc de moyenne nulle et de variance constante σ^2. Bien que le bruit blanc ne soit pas un modèle réaliste du signal de parole, sa simplicité permet d'obtenir des expressions analytiques assez simples et d'avoir un aperçu systématique et exact (c'est-à-dire sans erreurs d'estimation) du comportement de notre méthode de déréverbération en fonction des paramètres L_c et D.

4.1.1 Indicateurs de performance

Les indices de qualité que nous utilisons dans cette section sont les suivants :

$$\mathrm{SNR}_i = \frac{\mathbb{E}[(d(n))^2]}{\mathbb{E}[(r(n))^2]},$$

$$\mathrm{SNR}_f = \frac{\mathbb{E}[(d(n))^2]}{\mathbb{E}[(\hat{r}(n) - r(n))^2]},$$

$$\mathrm{SNRI} = \frac{\mathbb{E}[(r(n))^2]}{\mathbb{E}[(\hat{r}(n) - r(n))^2]}.$$

Pour un signal de bruit blanc, une dérivation analytique de ces quantités est possible. Plus précisément, on montre dans l'annexe A que

$$\mathrm{SNR}_i = \frac{||h_1||^2}{||h_2||^2},$$

$$\mathrm{SNR}_f = \frac{||h_1||^2}{\mathbb{E}[(\hat{r}(n) - r(n))^2]},$$

$$\mathrm{SNRI} = \frac{||h_2||^2}{\mathbb{E}[(\hat{r}(n) - r(n))^2]},$$

et

$$\mathbb{E}[(\hat{r}(n) - r(n))^2] = \hat{C}^T A \hat{C} + \sigma^2 ||h_2||^2 - 2\sigma^2 \sum_{k=0}^{L_c-1} \hat{C}_k \; R_h(D+k),$$

où les expressions de $||h_1||^2$, $||h_2||^2$, $\hat{C}$, A et R_h sont données dans l'annexe A. Ces résultats sont utilisés dans les deux sous-sections suivantes. Notons que la valeur des indicateurs de performance est indépendante de σ.

4.1.2 Performance de l'estimateur pour deux réponses impulsionnelles typiques

Notre partenaire nous a fourni plusieurs exemples de RI pertinentes dans le contexte de leur application (figures 4.1 et 4.4).

La RI numéro 1 est présentée dans la figure 4.1. Nous avons choisi $D = 160$ car l'intervalle $0 \leq k \leq D$ contient la partie structurée de la RI, c'est-à-dire les pics principaux. Cette RI satisfait $L_h = 1600$, c'est-à-dire que $h(k) \approx 0$ pour $k > 1600$, ainsi que $||h_1||^2/||h_2||^2 = 3,6$ et donc le signal désiré est 3,6 fois plus grand que la réverbération. Le SNRI et le SNR_f sont présentés en fonction de la longueur L_c du filtre dans les figures 4.2 et 4.3 respectivement. On observe que le SNRI et le SNR_f augmentent d'une façon monotone si L_c augmente, et plafonne aux environs de $L_c = L_h = 1600$. Pour $L_c = L_h$, on a SNRI $\approx 2,2$ et $\mathrm{SNR}_f \approx 8$, ce qui est une amélioration significative de la qualité du signal.

La RI numéro 4 est présentée dans la figure 4.4. Nous avons choisi $D = 100$ car l'intervalle $0 \leq k \leq D$ contient la partie structurée de la RI, c'est-à-dire les pics principaux. Cette RI satisfait $L_h = 1600$ ainsi que $||h_1||^2/||h_2||^2 = 4,9$ et donc le signal désiré est 4,9 plus grand que la réverbération. Le SNRI et le SNR_f sont présentés en fonction de la longueur L_c du filtre dans les figures 4.5 et 4.6 respectivement. On trouve le comportement monotone croissant des courbes précédentes. Les SNR se stabilisent aussi quand L_c est comparable à L_h. Pour $L_c = L_h$, on a SNRI $\approx 4,5$ et $\mathrm{SNR}_f \approx 22$, ce qui est une amélioration significative de la qualité du signal.

Pour examiner l'effet du paramètre D sur la performance, l'expérience précédente avec RI-4 a été répétée en utilisant $D = 60$ au lieu de $D = 100$, donc en ne gardant que les trois premiers pics de la RI. Dans ce cas, on a $\text{SNR}_i = ||h_1||^2/||h_2||^2 = 2,08$ et donc le signal désiré n'est que deux fois supérieur à la réverbération. La RI, le SNRI et le SNR_f sont représentés dans les figures 4.7, 4.8 et 4.9 respectivement. On retrouve encore le comportement monotone croissant. Pour $L_c = L_h$, on a SNRI $\approx 2,5$ et $\text{SNR}_f = 5$. Ces deux résultats sont inférieurs à ceux obtenus avec $D = 100$, mais par définition le signal désiré contient moins de réverbération puisque $D = 60 < 100$.

Pour les exemples considérés ici, on constate que 80% du SNRI peut être obtenu avec une valeur de L_c qui se situe entre 500 et 1000, soit approximativement entre $(1/3)L_h$ et $(2/3)L_h$. L'algorithme de débruitage a un coût de calcul qui augmente avec L_c. Pour un signal source de bruit blanc et pour les RI considérées, un compromis entre le coût de calcul et la qualité du débruitage peut être obtenu en choisissant une valeur de L_c qui se situe dans l'intervalle

$$\frac{1}{3}L_h \leq L_c \leq \frac{3}{4}L_h.$$

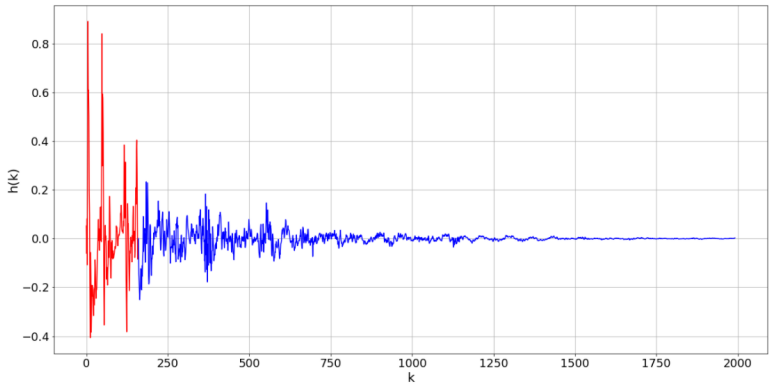

Figure 4.1 RI1 : La partie en rouge correspond à $k \leq D$ avec D=160.

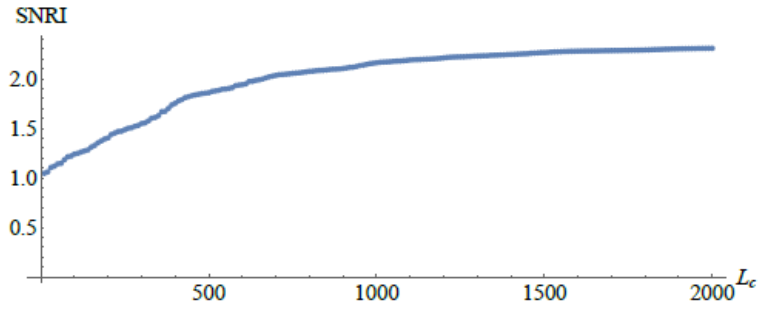

Figure 4.2 SNRI versus L_c pour la RI-1 avec D=160, pour une source de bruit blanc.

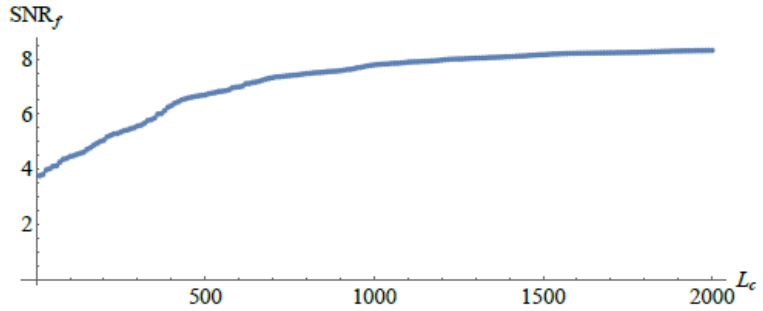

Figure 4.3 SNR_f versus L_c pour la RI-1 avec D=160, pour une source de bruit blanc.

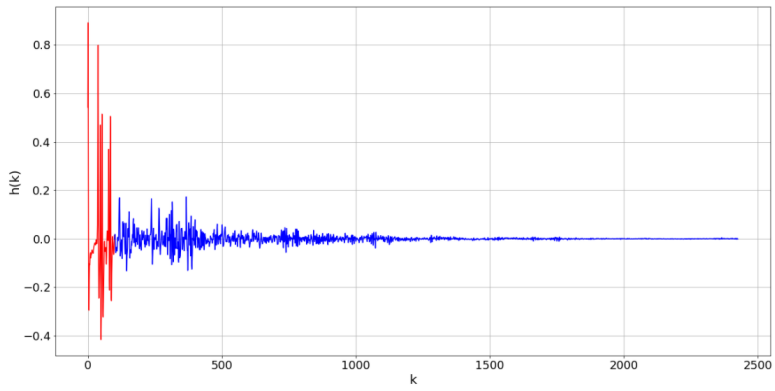

Figure 4.4 RI-4 : La partie en rouge correspond à $k \leq D$ avec D=100.

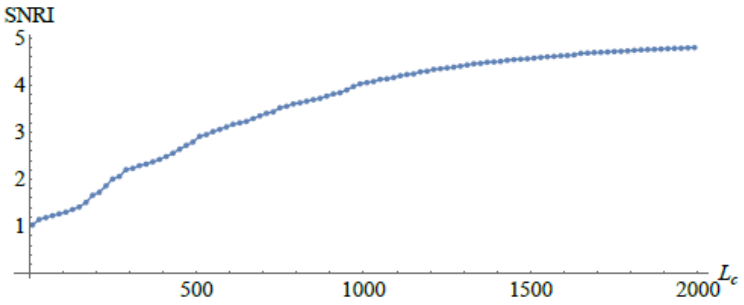

Figure 4.5 SNRI versus L_c pour la RI-4 avec D=100, pour une source de bruit blanc.

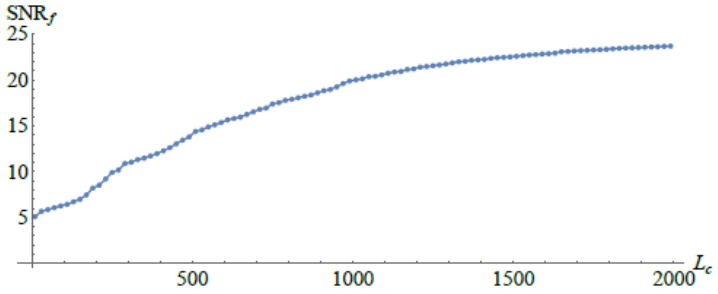

Figure 4.6 SNR_f versus L_c pour la RI-4 avec D=100, pour une source de bruit blanc.

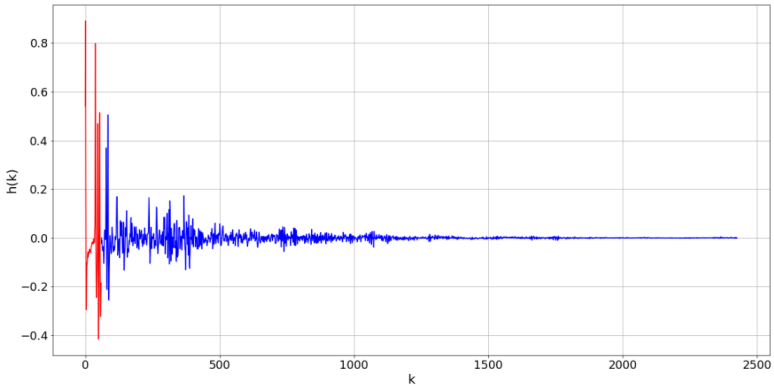

Figure 4.7 RI-4 : La partie en rouge correspond à $k \leq D$ avec D=60.

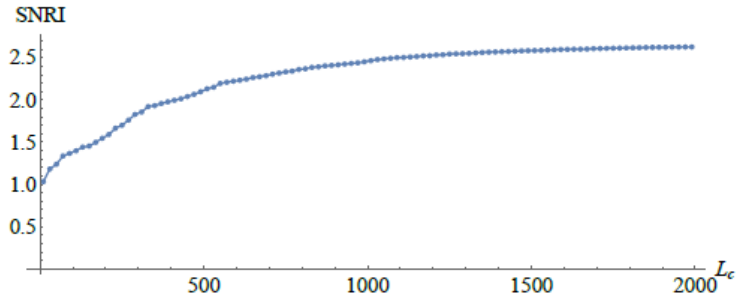

Figure 4.8 SNRI versus L_c pour la RI-4 avec D=60, pour une source de bruit blanc.

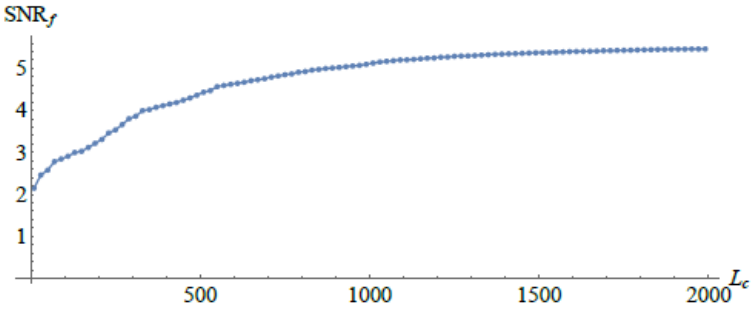

Figure 4.9 SNR_f versus L_c pour la RI-4 avec D=60, pour une source de bruit blanc.

4.1.3 Variabilité des indicateurs de performance en fonction de la partie d'apparence aléatoire de la réponse impulsionnelle

Une RI est typiquement composée d'un premier intervalle $0 \leq k \leq D$ où la fonction a quelques pics bien définis, suivi pour $k > D$ d'une autre partie d'apparence aléatoire. L'objectif de cette section est d'examiner la sensibilité de notre estimateur à la partie désordonnée de la RI.

Pour ce faire, nous construisons des RI synthétiques dont la première partie sera égale à RI-1, mais dont la seconde partie, donc $k > D$, sera composée d'un bruit aléatoire synthétique aux propriétés semblables à RI-1. Plus précisément, chaque réalisation h^* de la RI est définie par

$$h^*(i) = \begin{cases} h(i), & i \leq D-1, \\ \sigma e^{-(i-D)/(2.5D)} U_i, & i \geq D, \end{cases} \tag{4.1}$$

où h est la RI-1 avec $D = 160$ et les U_i sont obtenus en effectuant une moyenne mobile sur 4 points d'un bruit blanc gaussien de moyenne nulle et d'écart type $\sigma = 0,2$. Cette moyenne mobile introduit une corrélation des $h^*(i)$ qui permet de reproduire plus fidèlement les statistiques de la RI-1 dans sa partie d'apparence aléatoire. On peut constater la similarité entre la RI-1 et les RI synthétiques dans les figures 4.10 et 4.11, où la RI-1 et une RI synthétique typique sont représentées.

Les SNRI et SNR_f obtenus avec 5 réalisations sont représentés dans les figures 4.12 et 4.13. On voit que trois des cinq courbes atteignent un maximum pour $L_c = 1000$, tandis que deux autres continuent à augmenter avec L_c. On constate que la performance de débruitage peut être assez sensible à la structure de la partie irrégulière de la RI.

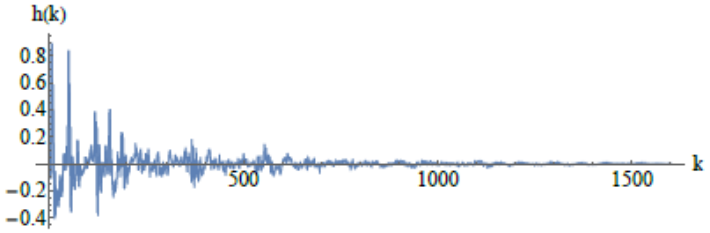

Figure 4.10 RI-1.

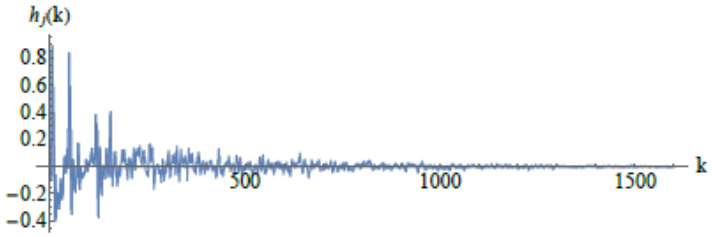

Figure 4.11 Une réalisation d'une variation synthétique de RI-1.

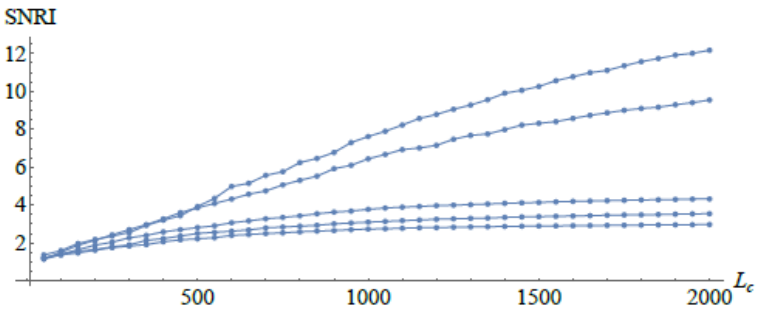

Figure 4.12 SNRI en fonction de L_c pour 5 réalisations de la RI synthétique.

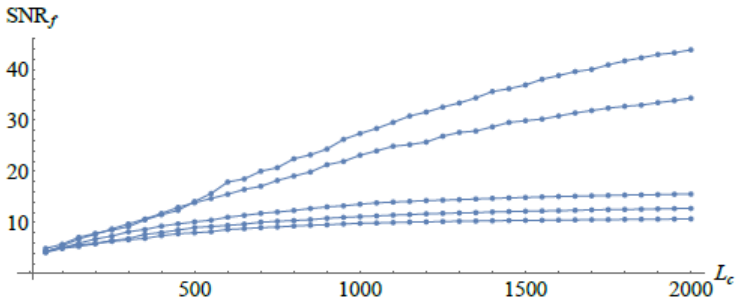

Figure 4.13 SNR_f en fonction de L_c pour 5 réalisations de la RI synthétique.

4.2 Application du pré-blanchiment du signal observé

Pour examiner l'effet du pré-blanchiment sur la performance de l'algorithme de déréverbération, on a appliqué la méthode de base décrite dans la section 3.3 avec les signaux de commande et RI-1. Le pré-blanchiment du signal observé a été fait par fenêtre de taille $L_f = 512$. On a fixé $L_c = 800$ et $D = 160$. Les deux tableaux 4.1 et 4.2 montrent les valeurs des indicateurs de performance pour deux signaux de paroles avec et sans pré-blanchiment. Dans le tableau 4.1, les valeurs de SNRI montrent qu'il n'y a aucune amélioration de la qualité du signal. Cependant, le tableau 4.2 montre qu'on a une bonne amélioration de la qualité de deux signaux si on n'applique pas le pré-blanchiment. De plus, les valeurs de PESQ pour les deux signaux avec blanchiment sont inférieures aux valeurs de PESQ obtenues pour les mêmes signaux sans blanchiment. Alors, on ne va pas considérer le pré-blanchiment dans nos prochains tests.

Tableau 4.1 Les mesures de performance de la méthode de base en appliquant le pré-blanchiment sur deux signaux de commande

Signal	PESQ	SNR_i	SNR_f	SNRI
Start coffee machine	1.42	3.56	3.61	1.01
Stop coffee machine	1.71	6.44	6.50	1.01

Tableau 4.2 Les mesures de performance de la méthode de base sans appliquer le pré-blanchiment sur deux signaux de commande

Signal	PESQ	SNR_i	SNR_f	SNRI
Start coffee machine	1.90	3.56	4.61	1.29
Stop coffee machine	2.10	6.44	10.46	1.62

4.3 Qualité de débruitage pour des signaux de parole supposés stationnaires

4.3.1 Description de l'expérience

Dans cette section, nous estimons les indicateurs de qualité de déréverbération pour deux sources de parole. La première est un extrait d'entrevue de durée 20 s. La seconde a été obtenue en juxtaposant huit signaux de commandes vocales pour une durée totale de 16 s. Dans ce cas, chacun des huit signaux de commande a été centré et réduit individuellement avant la concaténation pour rendre le signal global plus stationnaire. On caractérise le signal x mesuré au micro par sa fonction d'autocorrélation $R_x(\ell) := \mathbb{E}\left[x(n)\ x(n+\ell)\right]$ qu'on suppose indépendante de n, ce qui correspond à une hypothèse de stationnarité faible. Cette expérience correspond à un contexte d'application où les locuteurs sont toujours dans la même pièce et situés au même endroit dans cette pièce, de sorte que la réponse impulsionnelle associée à la réverbération est constante. On calcule le filtre de déréverbération avec toutes les données disponibles. Pour chacun des deux signaux sources, on applique le filtre globalement et on calcule SNRI et SNR_f en fonction de la longueur du filtre L_c. On applique ensuite le filtre localement, c'est à dire à des parties du signal source, pour tester la performance locale d'un filtre optimisé globalement.

Les indicateurs de performance définis à la section 4.1.1 sont entièrement spécifiés par les quantités $\mathbb{E}\left[(d(n))^2\right]$, $\mathbb{E}\left[(r(n))^2\right]$ et $\mathbb{E}\left[(\hat{r}(n)-r(n))^2\right]$. Le signal de réverbération estimé $\hat{r}$ est calculé avec $\hat{r}(n) = \sum_{k=0}^{L_c-1} \hat{C}(k)\, x(n-k)$ avec le filtre $\hat{C}$ obtenu en résolvant l'équation (A.2), où la matrice $A \in \mathbb{R}^{L_c \times L_c}$ et le vecteur $B \in \mathbb{R}^{L_c}$ sont obtenus avec les équations (A.3) et (A.4), en utilisant l'estimateur

$$\hat{R}_x(\ell) := \frac{1}{N} \sum_{n=0}^{N-\ell} x(n)\, x(n+\ell) \tag{4.2}$$

pour la fonction d'autocorrélation $R_x(\ell)$. Le vecteur x est centré avant cette estimation. L'estimateur (4.2) est asymptotiquement non biaisé dans la limite $N \to \infty$.

Les signaux x, d et r sont obtenus avec les produits de convolution $x = h * s$, $d = h_1 * s$ et $r = h_2 * s$, où $h_1 := (h(0),\, h(1),\, ...,\, h(D-1), \underbrace{0,\, 0,\, ...,\, 0}_{L_h - D \text{ zéros}})^T$ et
$h_2 := (\underbrace{0,\, 0,\, ...,\, 0}_{D \text{ zéros}},\, h(D),\, h(D+1),\, ...,\, h(L_h-1))^T$. Pour estimer les quantités $\mathbb{E}\left[(d(n))^2\right]$, $\mathbb{E}\left[(r(n))^2\right]$ et $\mathbb{E}\left[(\hat{r}(n)-r(n))^2\right]$, on utilise les estimateurs

$$\frac{1}{N}\sum_{n=0}^{N-1}(d(n))^2, \quad \frac{1}{N}\sum_{n=0}^{N-1}(r(n))^2 \quad \text{et} \quad \frac{1}{N}\sum_{n=0}^{N-1}(r(n)-\hat{r}(n))^2.$$

4.3.2 SNRI et SNR_f versus L_c pour RI-1

Le SNRI et le SNR_f obtenus pour le signal d'entrevue complet sont représentés dans les figures 4.14 et 4.15. On constate que SNRI et SNR_f plafonnent pour $L_c \geq 500$. Le SNRI atteint une valeur maximale d'environ 1,6, ce qui est une amélioration significative de la qualité du signal. Le débruitage nous fait passer d'un SNR_i de 3,9 à un SNR_f de 6,3. L'histogramme des SNRI obtenus avec $L_c = 500$ pour des échantillons de taille 10000 points est représenté dans la figure 4.16. Nous avons considéré toutes les fenêtres de taille 10000 distantes les unes des autres de 5000 points. Soulignons que la taille de 10000 a été choisie pour être comparable à celle des signaux de commande, dont la taille varie entre 10000 et 20000 points. On voit que les SNRI locaux varient entre 1 et 2,5, et que la qualité du signal est améliorée (ou gardée constante) pour tous les échantillons. Ce résultat montre qu'un filtre qui n'est pas adapté au signal localement peu néanmoins améliorer la qualité du signal systématiquement.

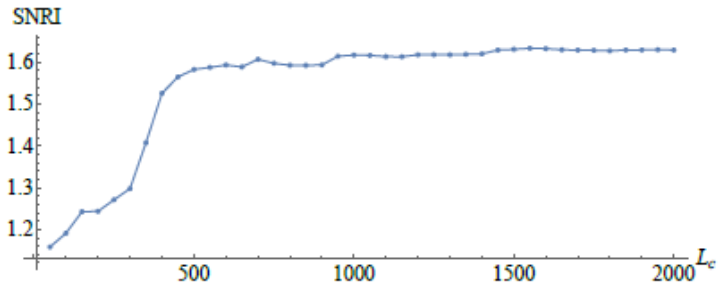

Figure 4.14 SNRI versus L_c pour le signal d'entrevue avec RI-1 et $D = 160$ ($\text{SNR}_i = 2,9$).

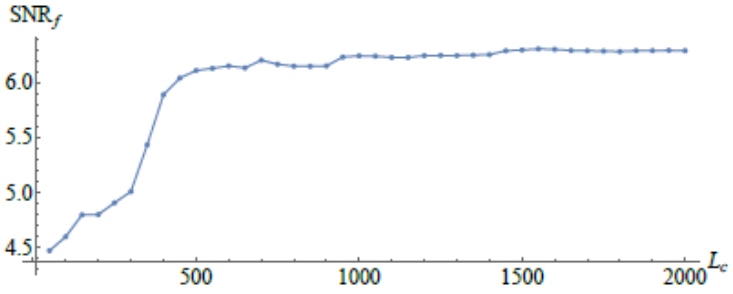

Figure 4.15 SNR_f versus L_c pour le signal d'entrevue avec RI-1 et $D = 160$ ($\text{SNR}_i = 2,9$).

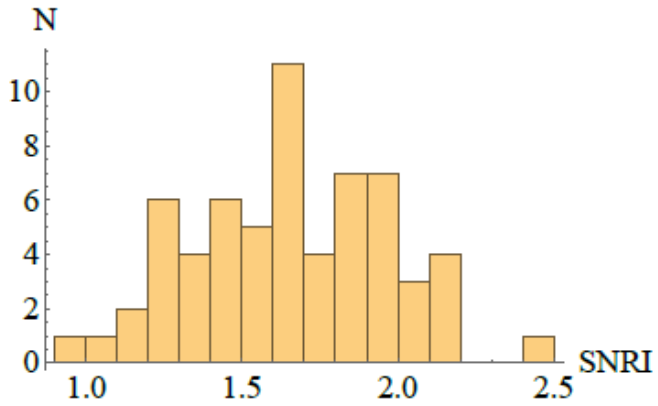

Figure 4.16 Histogramme des SNRI calculés pour des fenêtres de 10000 points avec $L_c = 500$, RI-1 et $D = 160$, pour le signal d'entrevue.

Le SNRI et le SNR_f obtenus pour le signal de commandes agglomérées complet sont représentés dans les figures 4.17 et 4.18. On constate que les SNR plafonnent pour $L_c \geq 500$. Le SNRI atteint une valeur maximale d'environ 1,45, ce qui est une amélioration significative de la qualité du signal. Le débruitage nous fait passer d'un SNR_i de 2,9 à un SNR_f de 4,3. L'histogramme des SNRI obtenus avec $L_c = 500$ pour des échantillons de taille 10000 points est représenté dans la figure 4.19. On voit que les SNRI locaux varient entre 0,5 et 4,0. La qualité du signal est améliorée (ou gardée constante) pour environ 80% des échantillons (c'est à dire SNRI ≥ 1), et empirée pour 20% (c'est à dire SNRI < 1).

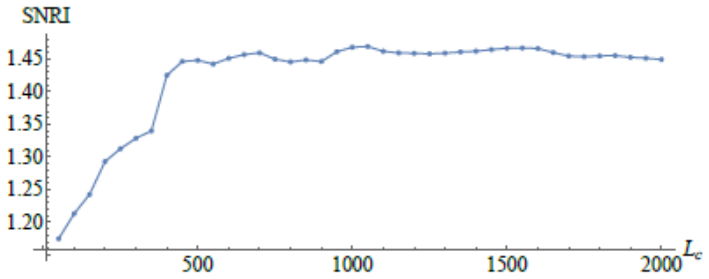

Figure 4.17 SNRI versus L_c pour le signal de commandes avec RI-1 et $D = 160$ ($\text{SNR}_i = 2,9$).

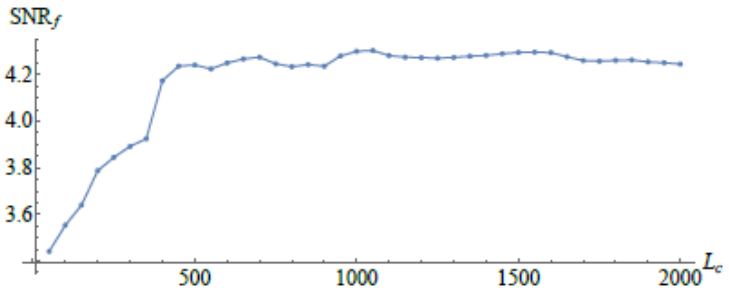

Figure 4.18 SNR_f versus L_c pour le signal de commandes avec RI-1 et $D = 160$ ($\text{SNR}_i = 2,9$).

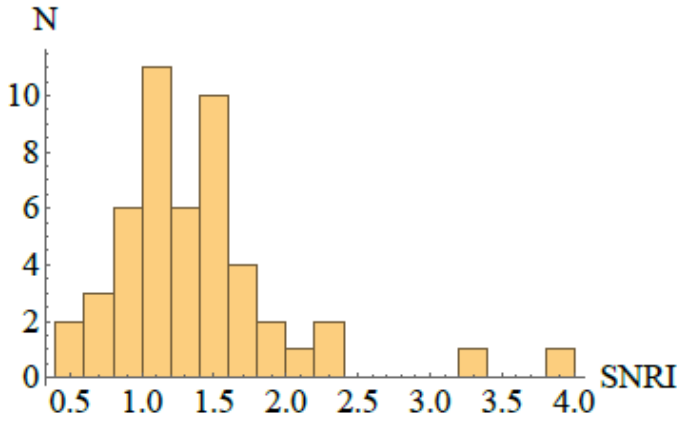

Figure 4.19 Histogramme des SNRI calculés pour des fenêtres de 10000 points avec $L_c = 500$, RI-1 et $D = 160$, pour le signal de commandes.

4.3.3 SNRI et SNR_f versus L_c pour RI-4

Le SNRI et le SNR_f obtenus pour le signal d'entrevue complet sont représentés dans les figures 4.20 et 4.21. On constate que les SNR plafonnent pour $L_c \geq 1000$. Le SNRI atteint une valeur maximale d'environ 1,03, ce qui n'est pas une amélioration significative de la qualité du signal. L'histogramme des SNRI obtenus avec $L_c = 1000$ pour des échantillons de taille 10000 points est représenté dans la figure 4.22. On voit que les SNRI locaux varient entre 0,3 et 1,6, et donc que la qualité du signal est empirée pour au moins la moitié des échantillons. Dans ce cas, le $\text{SNR}_i = 6,3$ est relativement élevé et donc il est plus difficile d'améliorer la qualité du signal, qui est relativement bonne au départ (par rapport aux résultats avec RI1, où on avait $\text{SNR}_i = 2,9$).

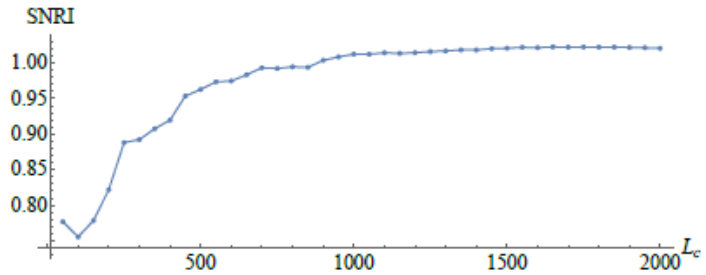

Figure 4.20 SNRI versus L_c pour le signal d'entrevue avec RI-4 et $D = 100$ ($\text{SNR}_i = 6,3$).

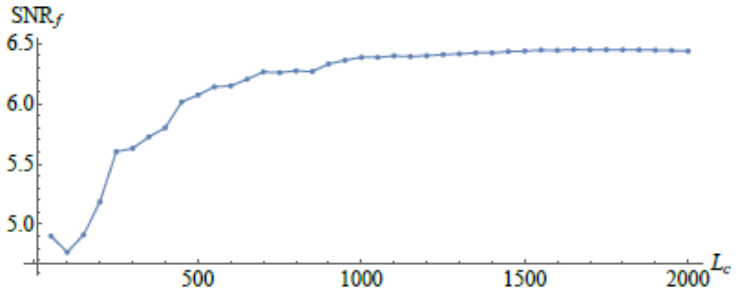

Figure 4.21 SNR_f versus L_c pour le signal d'entrevue avec RI-4 et $D = 100$ ($\text{SNR}_i = 6,3$).

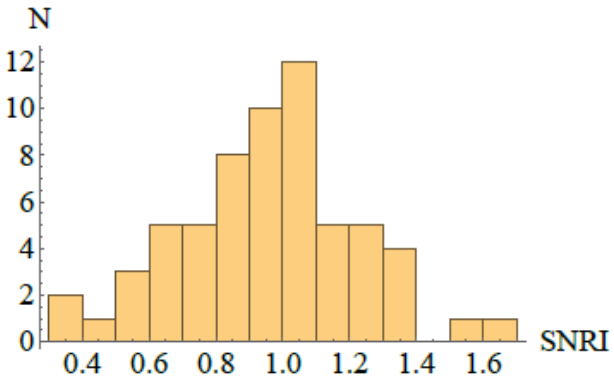

Figure 4.22 Histogramme des SNRI calculés pour des fenêtres de 10000 points avec $L_c = 1000$, RI-4 et $D = 100$, pour le signal d'entrevue.

Le SNRI et le SNR_f obtenus pour le signal de commandes sont représentés dans les figures 4.23 et 4.24. On constate que les SNR plafonnent pour $L_c \geq 500$. Le SNRI atteint une valeur maximale d'environ 1,37, ce qui est une amélioration significative de la qualité du signal. L'histogramme des SNRI obtenus avec $L_c = 500$ pour des échantillons de taille 10000 points est représenté dans la figure 4.25. On voit que les SNRI locaux varient entre 0,4 et 2,0. Ici la qualité du signal est améliorée pour environ 60% des échantillons, et empirée pour 40%. Ici les résultats sont un peu meilleurs que pour le signal d'entrevue car la valeur du SNR_i est un peu moins grande, soit 3,9.

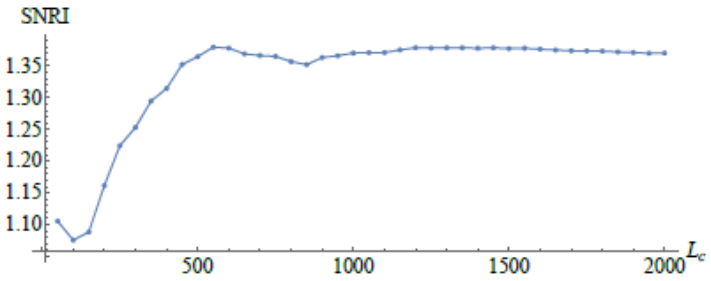

Figure 4.23 SNRI versus L_c pour le signal de commandes avec RI-4 et $D = 100$ ($\text{SNR}_i = 3,9$).

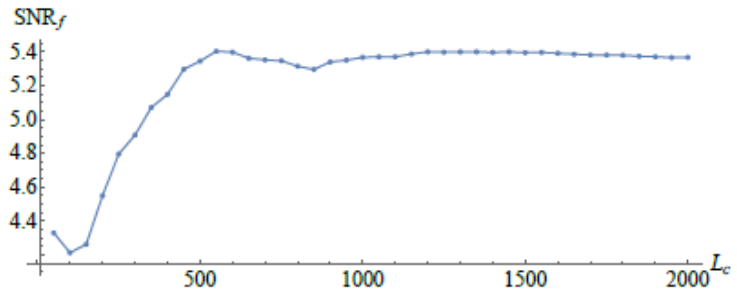

Figure 4.24 SNR$_f$ versus L_c pour le signal de commandes avec RI-4 et $D = 100$ (SNR$_i = 3, 9$).

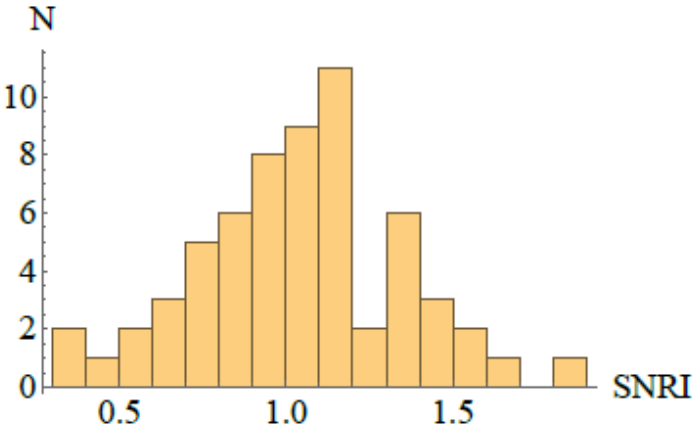

Figure 4.25 Histogramme des SNRI calculés pour des fenêtres de 10000 points avec $L_c = 500$, RI-4 et $D = 100$, pour le signal de commandes.

4.3.4 Un aperçu de la fonction d'autocorrélation $R_x(\ell)$, des signaux r, $\hat{r}$ et du filtre $\hat{C}$

Dans cette section, nous présentons quelques résultats intermédiaires obtenus pour le signal d'entrevue avec RI-1. La fonction d'autocorrélation $R_x(\ell)$ est estimée avec (4.2) sur des intervalles consécutifs disjoints de taille 10000 points. La fonction d'autocorrélation du ième intervalle est dénotée par $R_x^{(i)}(\ell)$. Pour chaque valeur de $\ell \in \{1, 2, ..., 5000\}$, on calcule l'estimé de $R_x(\ell)$, qui est

$$\hat{R}_x(\ell) := \frac{1}{m} \sum_{i=1}^{m} R_x^{(i)}(\ell),$$

où m est le nombre d'intervalles, et l'écart-type σ_ℓ des $R_x^{(i)}(\ell)$. En supposant les $R_x^{(i)}(\ell)$ indépendants, l'écart-type $\hat{\sigma}_\ell$ de l'estimé $\hat{R}_x(\ell)$ est donné par $\hat{\sigma}_\ell = \sigma_\ell/\sqrt{m}$. Dans la figure 4.26, on représente la fonction $\hat{R}_x(\ell)$ obtenue avec le signal d'entrevue et la RI-1, ainsi que les courbes $\hat{R}_x(\ell) \pm 2\,\hat{\sigma}_\ell$ (les deux courbes plus pâles). Pour les ℓ pour lesquels $|\hat{R}_x(\ell)|$ devient petit par rapport à $2\,\hat{\sigma}_\ell$, donc pour $\ell \geq 4500$, on pose $\hat{R}_x(\ell) = 0$. L'estimé final utilisé est représenté à la figure 4.27.

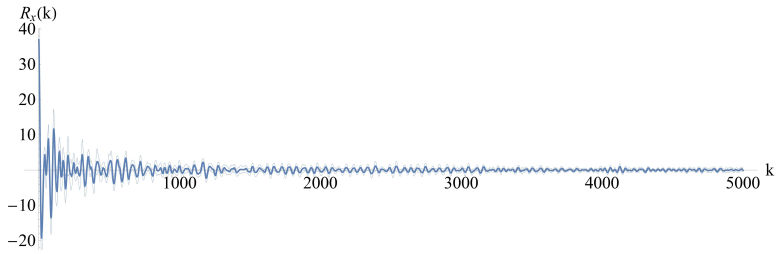

Figure 4.26 Fonction $\hat{R}_x(\ell)$ (courbe fonçée) et fonctions $\hat{R}_x(\ell) \pm 2\hat{\sigma}_\ell$ (les deux courbes pâles).

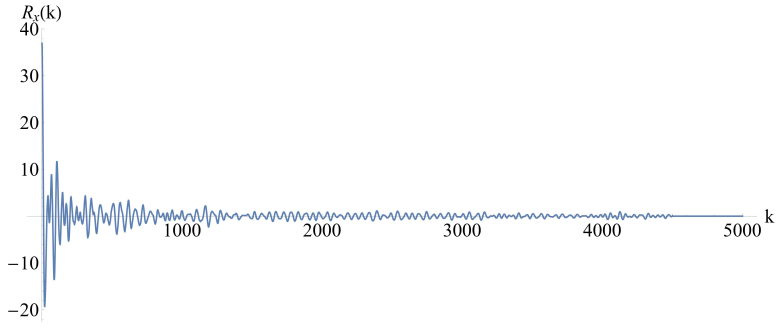

Figure 4.27 Fonction $\hat{R}_x(\ell)$ aprés annulation des $\hat{R}_x(\ell)$ pour $\ell \geq 4500$.

Dans la figure 4.28, on montre les signaux $r(n)$ et $\hat{r}(n)$ côte-à-côte. On voit que la corrélation entre $\hat{r}(n)$ et $r(n)$ existe mais elle n'est pas visible en tout point. Le graphe de $\hat{r}(n)$ versus $r(n)$, représenté dans la figure 4.29, met en évidence cette corrélation imparfaite. Le filtre $\hat{C}$ obtenu pour le signal d'entrevue et RI-1 est représenté dans la figure 4.30. On observe une augmentation de son amplitude à l'extrémité droite. Cette fluctuation est liée au fait que le modèle $r = C^T x$ ne permet pas, en général, une représentation exacte du signal de réverbération.

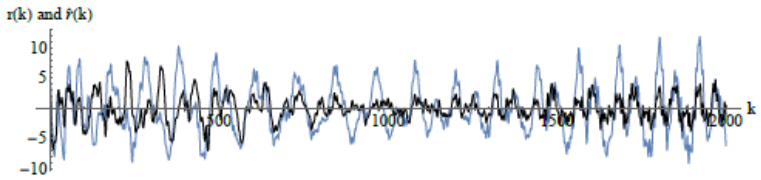

Figure 4.28 Les signaux $r(n)$ (en bleu) et $\hat{r}(n)$ (en noir).

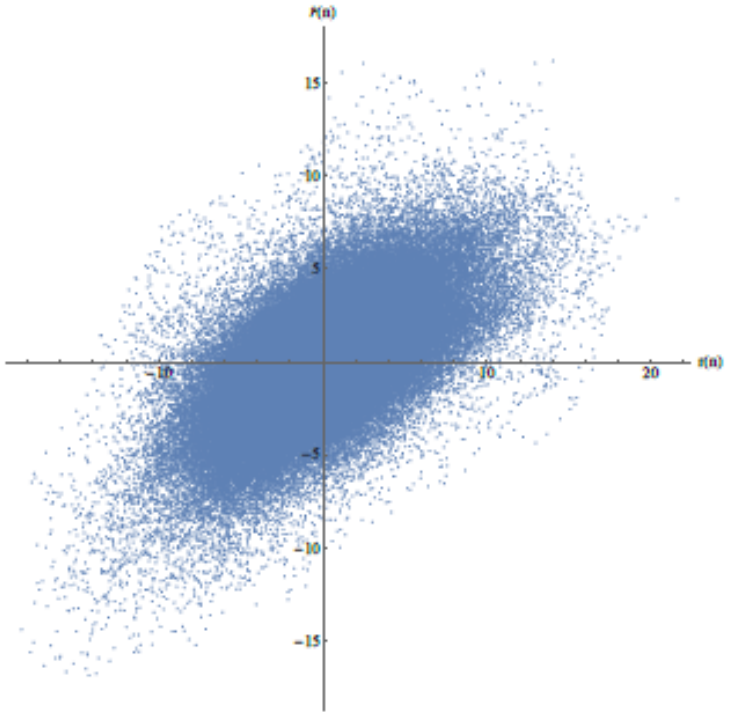

Figure 4.29 $r(n)$ versus $\hat{r}(n)$.

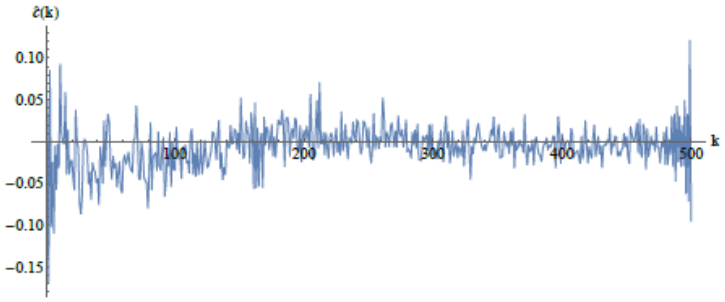

Figure 4.30 Filtre $\hat{C}$ obtenu pour le signal d'entrevue et RI-1.

4.4 Performance de la méthode de base pour les signaux de commande

Dans cette section, on présente les valeurs des indicateurs de performance obtenues pour les signaux de commande avec RI-1 et RI-4 représentées dans les figures 4.1 et 4.4. Dans la section 4.1.2, on a donné une explication pour le choix de la longueur L_c du filtre et le décalage D. Pour RI-1, on choisit $D = 160$ et $D = 100$ pour RI-4 . Pour la longueur L_c du filtre, on a montré que $L_c = 500$ est un bon compromis entre précision et temps de calcul. Les signaux de commande utilisés dans la suite sont représentés dans le tableau 4.3.

Tableau 4.3 Les signaux de commande

Numéro du signal	Contenu du signal	Durée du signal
Signal 1	Start coffee machine	1,36 s
Signal 2	Stop coffee machine	1,44 s
Signal 3	Open the gas tank	1,74 s
Signal 4	Tune in traffic information	2,02 s
Signal 5	Why is the maintenance light on	3,07 s
Signal 6	Turn on home security	2,51 s
Signal 7	Turn off vacation settings	2,14 s
Signal 8	Open the hood	1,75 s

L'algorithme implémenté pour estimer le signal désiré à partir de la version réverbérée de chacun des signaux de commande est présenté dans la section 3.3. La méthode de base consiste à calculer d'abord la variance locale du signal observé par une moyenne mobile avec une fenêtre de taille $L_f = 512$. Le choix de la taille de la fenêtre a été fait en se basant sur l'idée que le signal de la parole est approximativement stationnaire pour des périodes de 20 ms à 30 ms [2]. Puisqu'on utilise une fréquence d'échantillonnage de 16 kHz, alors 30 ms correspond à 480 points. De plus, notre partenaire nous a proposé de considérer des fenêtres de taille 512 points car leur système fait le traitement sur des trames de cette taille. Le calcul

de A, B et $\hat{C}$ se fait pour tout le signal en utilisant les équations (3.14), (3.15) et (3.16) respectivement. Le débruitage du signal réverbéré se fait à la fin selon l'équation (3.17). Le nombre d'itérations utilisé dans nos tests est égal à 1, car utiliser plus d'une seule itération n'améliore pas significativement la qualité des résultats et augmente le temps de calcul. On désignera algorithme décrit ci-dessus par *méthode de base* dans la suite de ce chapitre

Dans la suite de ce chapitre, on fixe $D = 160$ pour RI-1 et $D = 100$ pour RI-4. Pour ce test, on fixe $L_c = 500$. Les indicateurs de performance obtenus pour les signaux de commande avec la RI-1 sont représentés dans le tableau 4.4. En regardant les valeurs de SNR_f et SNRI pour tous les signaux, on peut dire qu'il y a une amélioration significative de la qualité du signal. Par exemple, pour le signal 6, on a $\text{SNR}_i = 1,77$, cela veut dire que le signal désiré est presque 2 fois plus grand que le signal réverbéré. En appliquant l'algorithme de déréverbération, on a obtenu un $\text{SNR}_f = 6,10$, c'est-à-dire que le signal désiré est devenu 6 fois plus grand que le signal réverbéré. Cette amélioration s'exprime par la valeur de SNRI qui est égale à 3,31. Dans la section 4.1.2, la valeur de SNRI avec RI-1 et $L_c = 500$ est 1,9. Dans le tableau 4.4, la valeur moyenne des SNRI pour tous les signaux est de 1,65. Dans la section 4.3.2, pour le signal de commandes agglomérées et RI-1, on avait obtenu un SNRI global un peu inférieur, soit 1,45 (figure 4.17), et les SNRI locaux (figure 4.19) étaient inférieurs à 1 pour 25% des intervalles. Par contre, dans le tableau 4.4, on voit que les SNRI sont tous supérieurs à 1,07. Ceci montre que le filtre calculé localement avec la méthode de base est plus performant que le filtre global utilisé dans la section 4.3. On peut aussi dire qu'on obtient déjà 87% de l'amélioration qu'on peut obtenir pour un signal source de bruit blanc. De plus, les valeurs de PESQ, qui sont entre 2 et 3, montrent que nos signaux estimés ont une bonne qualité perceptuelle. Cela signifie que cet algorithme de déréverbération devrait pouvoir améliorer la performance du système de reconnaissance automatique de la parole de notre partenaire. Notons cependant que les temps de calcul requis sont loin du temps réel, qui serait l'idéal pour l'application visée.

On a répété le test précédent mais cette fois-ci avec RI-4 et $D = 100$. Les résultats obtenus sont présentés dans le tableau 4.5. On constate que le SNRI moyen est de 1,86 et que la meilleure valeur de SNRI obtenue est 2,62 pour le signal 2. Il y a 4 des 8 signaux qui ont un SNRI ≥ 2, ce qui est une amélioration significative de la qualité du signal. Les valeurs du PESQ, entre 2 et 3, correspondent aussi à une bonne qualité perceptuelle. Dans la section 4.3.2, pour le signal de commandes agglomérées et RI-4, on avait obtenu un SNRI global de 1,37 (figure 4.23) et les SNRI locaux étaient inférieurs à 1 pour la moitié des intervalles. Ceci confirme que le filtre calculé localement performe nettement mieux que le filtre global obtenu à partir de signaux plus longs.

Pour tester l'effet de la normalisation par une variance locale par fenêtre sur la performance

Tableau 4.4 Les mesures de performance de la méthode de base pour les signaux de commande et RI-1

Signal	PESQ	SNR_i	SNR_f	SNRI	Temps de calcul
Signal 1	2,03	3,56	4,48	1,26	1 min 45 s
Signal 2	2,38	6,44	10,11	1,57	1 min 50 s
Signal 3	2,37	5,41	5,82	1,07	2 min 11 s
Signal 4	2,62	2,62	3,39	1,29	2 min 33 s
Signal 5	2,35	2,25	3,36	1,49	3 min 52 s
Signal 6	2,46	1,77	5,87	3,31	3 min 8 s
Signal 7	2,44	3,22	6,10	1,89	2 min 42 s
Signal 8	2,39	3,95	5,29	1,34	2 min 16 s

Tableau 4.5 Les mesures de performance de la méthode de base pour les signaux de commande et RI-4

Signal	PESQ	SNR_i	SNR_f	SNRI	Temps de calcul
Signal 1	2,32	2,01	4,62	2,30	1 min 44 s
Signal 2	2,30	2,32	6,08	2,62	1 min 49 s
Signal 3	2,31	13,97	20,06	1,43	2 min 18 s
Signal 4	2,69	4,13	7,86	1,90	2 min 38 s
Signal 5	2,54	3,89	7,79	2,00	4 min 6 s
Signal 6	2,26	4,12	8,64	2,05	3 min 11 s
Signal 7	2,39	3,43	4,25	1,24	2 min 49 s
Signal 8	2,51	3,38	4,61	1,36	2 min 10 s

de la méthode, on a utilisé une variance constante sur tout le signal et on a refait les mêmes tests avec RI-1 et RI-4. Les valeurs des indicateurs de performance sont représentées dans le tableau 4.6. En comparant les valeurs de SNRI dans le tableau 4.6 avec celles du tableau 4.4, on constate que les 5 premiers signaux du tableau 4.6 ont des valeurs de SNRI plus grandes que dans le tableau 4.4. Par exemple, pour le signal 1, sa valeur de SNRI est passée de 1,26 à 1,35. De même, pour le signal 4, la différence entre les deux valeurs de SNRI est de 0,15, ce qui est une amélioration. Par contre, pour la mesure PESQ, les valeurs de PESQ représentées dans le tableau 4.6 sont significativement plus petites que les valeurs obtenues dans le tableau 4.4 puisqu'elles sont presque toutes inférieures à 2.

On a aussi testé l'effet de la normalisation par une variance constante sur tous les signaux de commande avec RI-4. Le tableau 4.7 représente les valeurs des indicateurs de performance. En comparant les deux tableaux 4.7 et 4.5, on constate que seuls les signaux 2 et 4 ont des valeurs de SNRI que grandes que celles trouvées avec la normalisation par une variance locale. Pour le signal 5, la valeur de SNRI est passée de 2 à 1,59, ce qui est une différence significative. On remarque aussi que les valeurs de PESQ sont toutes inférieures aux valeurs

Tableau 4.6 Les mesures de performance de la méthode de base avec une variance constante pour les signaux de commande et RI-1

Signal	PESQ	SNR_i	SNR_f	SNRI	Temps de calcul
Signal 1	1,55	3,56	4,82	1,35	1 min 44 s
Signal 2	2,30	6,44	10,53	1,63	1 min 49 s
Signal 3	2,00	5,41	6,00	1,11	2 min 11 s
Signal 4	1,60	2,62	3,77	1,44	2 min 34 s
Signal 5	1,86	2,25	3,59	1,59	3 min 54 s
Signal 6	1,59	1,77	5,61	3,17	3 min 13 s
Signal 7	1,70	3,22	5,68	1,76	2 min 42 s
Signal 8	1,66	3,95	4,65	1,18	2 min 12 s

trouvées avec la normalisation par une variance locale.
L'utilisation d'une variance constante ne conduit pas toujours à un SNRI plus grand, mais elle provoque toujours une baisse du PESQ. Comme le PESQ reflète la clarté perceptuelle du signal, ces résultats suggèrent que la normalisation par une variance locale est préférable.

Tableau 4.7 Les mesures de performance de la méthode de base avec une variance constante pour les signaux de commande et RI-4

Signal	PESQ	SNR_i	SNR_f	SNRI	Temps de calcul
Signal 1	1,60	2,01	4,12	2,05	1 min 43 s
Signal 2	2,13	2,32	6,71	2,89	2 min 2 s
Signal 3	1,64	13,97	18,06	1,29	2 min 16 s
Signal 4	1,47	4,13	7,93	1,92	2 min 39 s
Signal 5	2,19	3,89	6,20	1,59	4 min 3 s
Signal 6	1,78	4,21	7,75	1,84	3 min 9 s
Signal 7	1,75	3,43	4,20	1,22	2 min 47 s
Signal 8	2,07	3,38	4,58	1,35	2 min 9 s

4.5 Application de la méthode de base par fenêtre et débruitage à la fin

La nécessité pratique de faire un traitement du signal par fenêtre nous conduit à évaluer la performance de plusieurs versions de l'algorithme de base. Nous considérons ici la version où la variance du signal est supposée constante dans chaque fenêtre de taille $L_f = 512$. La taille de la fenêtre limite la taille du filtre. En effet, la taille du filtre $\hat{C}$ doit satisfaire la condition $L_c + D \leq L_f$. Pour les tests présentés dans cette section, on a fixé $L_c = 128$. Le calcul de

A et B se fait par fenêtre en cumulant ses valeurs jusqu'à la fin du signal. En arrivant à la dernière fenêtre, on calcule le filtre $\hat{C}$ et on applique le débruitage du signal réverbéré.
Les résultats des indicateurs de performance pour l'implémentation de la méthode de base par fenêtre avec RI-1 et RI-4 sont présentés dans les tableaux 4.8 et 4.9. En réduisant la taille L_c du filtre de 800 à 128, on peut constater que le temps de calcul pour l'implémentation de l'algorithme par fenêtre est nettement plus petit. Par exemple, pour RI-1, le temps nécessaire de calcul pour le signal 5 qui dure 3,07 secondes est 38 secondes au lieu de 3 minutes et 52 secondes. La valeur du SNRI pour ce signal est passée de 1,49 à 1,23. Cette différence s'explique par la réduction de la taille L_c du filtre. En effet, on a montré dans la section 4.3.2 que le SNRI augmente quand L_c augmente pour le signal de commandes. La figure 4.17 montre que pour $L_c = 500$, on a SNRI=1,6 alors que pour $L_c = 128$ on a SNRI=1,2. Les valeurs de PESQ trouvées avec la méthode de base appliquée sur tout le signal sont comparables aux valeurs présentées dans le tableau 4.8. La même expérience a été faite avec RI-4. Dans le tableau 4.9, on voit que les valeurs de SNRI sont plus petites que celles du tableau 4.5 mais restent toujours supérieur à 1,20 sauf pour les deux signaux 7 et 8. Les valeurs de PESQ restent comparables pour les deux implémentations de la méthode de base. Enfin, l'implémentation de la méthode de base par fenêtre nous a permis de gagner beaucoup au niveau du temps de calcul, au prix d'une baisse du SNRI mais avec des résultats comparables pour le PESQ. Les temps de calcul obtenus avec la méthode de base ou la méthode de base par fenêtre sont beaucoup plus grands que les attentes de notre partenaire, qui cherche une solution en temps réel avec un temps de latence qui ne dépasse pas 100 ms. Dans le but de réduire le temps de calcul, on présente dans la prochaine section les résultats obtenus avec l'implémentation rapide proposée dans la section 3.5.

Tableau 4.8 Les mesures de performance de la méthode de base implémentée par fenêtre pour les signaux de commande et RI-1

Signal	PESQ	SNR_i	SNR_f	SNRI	Temps de calcul
Signal 1	2,34	3,56	4,48	1,26	17,1 s
Signal 2	2,46	6,44	9,97	1,55	17,7 s
Signal 3	2,54	5,23	5,44	1,04	21,5 s
Signal 4	2,71	2,62	3,34	1,27	24,9 s
Signal 5	2,35	2,25	2,78	1,23	38,00 s
Signal 6	2,54	1,77	4,90	2,77	31,4 s
Signal 7	2,31	3,22	5,58	1,73	27,2 s
Signal 8	2,39	3,95	4,64	1,13	21,4 s

Tableau 4.9 Les mesures de performance de la méthode de base implémentée par fenêtre pour les signaux de commande et RI-4

Signal	PESQ	SNR_i	SNR_f	SNRI	Temps de calcul
Signal 1	2,40	2,01	4,19	2,09	16,6 s
Signal 2	2,55	2,32	4,47	1,93	19,5 s
Signal 3	2,59	13,97	17,66	1,26	21,7 s
Signal 4	2,80	4,13	5,23	1,27	25,2 s
Signal 5	2,58	3,89	4,97	1,27	39,00 s
Signal 6	2,53	4,21	5,66	1,34	32,8 s
Signal 7	2,41	3,43	3,64	1,06	26,8 s
Signal 8	2,49	3,38	3,86	1,14	21,6 s

4.6 Application de la méthode rapide par fenêtre et débruitage à la fin

Dans cette partie, notre objectif est d'évaluer la performance de la méthode rapide par fenêtre présentée dans la section 3.5. La fonction d'autocorrélation calculée dans une fenêtre est normalisée par la variance de la fenêtre courante. Les fonctions d'autocorrélations ainsi normalisées sont cumulées jusqu'à la fin du signal. À la fin, le calcul de A et B se fait en utilisant les équations (3.32) et (3.35) respectivement. Le débruitage du signal observé x se fait en utilisant le filtre $\hat{C}$ calculé pour tout le signal.

Les résultats des indicateurs de performance pour RI-1 et RI-4 sont présentés respectivement dans les tableaux 4.10 et 4.11. L'implémentation rapide nous a permis de réduire le temps de calcul d'un facteur 38, ce qui est significatif pour l'application de la déréverbération en temps réel. Pour RI-1, 7 sur 8 signaux ont des valeurs de SNRI qui varient entre 1,12 et 1,50. Seul le signal 6 a un SNRI supérieur à 2. Le signal 6 a un SNR_i inférieur à 2, alors il est plus facile d'améliorer la qualité du signal 6 car les autres signaux ont des SNR_i élevés. La distribution des valeurs de SNRI est comparable à celle trouvée avec le signal des commandes agglomérées présentées dans la figure 4.19. Les valeurs de PESQ correspondent aussi à une bonne qualité perceptuelle. Pour RI-4, la valeur maximale de SNRI obtenue est 2,03 pour le signal 1. Pour les autres signaux, les valeurs de SNRI varient entre 1,06 et 1,70. On constate que le signal 3 un SNR_i égal à 13,97. Même si cette valeur est élevée, l'algorithme de déréverbération a amélioré la qualité du signal. De plus, les valeurs de PESQ qui varient entre 2,49 et 2,87 montrent une bonne bonne qualité perceptuelle des signaux débruités. Notre partenaire souhaite avoir une solution implémentée en temps réel. Cela implique que le traitement de la déréverbération doit être fait sur chaque fenêtre dès qu'une nouvelle fenêtre de données devient disponible. On désignera cette implémentation par *implémentation rapide par fenêtre*. Une description détaillée de cette implémentation est présentée dans la prochaine section.

Tableau 4.10 Les mesures de performance de l'implémentation rapide pour les signaux de commande et RI-1

Signal	PESQ	SNR_i	SNR_f	SNRI	Temps de calcul
Signal 1	2,45	3,56	4,18	1,17	432 ms
Signal 2	2,72	6,44	8,63	1,34	480 ms
Signal 3	2,52	5,41	6,45	1,19	558 ms
Signal 4	2,89	2,62	3,34	1,27	668 ms
Signal 5	2,35	2,25	2,62	1,16	976 ms
Signal 6	2,51	1,77	3,80	2,14	774 ms
Signal 7	2,44	3,22	4,82	1,50	681 ms
Signal 8	2,69	3,95	4,42	1,12	550 ms

Tableau 4.11 Les mesures de performance de l'implémentation rapide pour les signaux de commande et RI-4

Signal	PESQ	SNR_i	SNR_f	SNRI	Temps de calcul
Signal 1	2,62	2,01	4,08	2,03	410 ms
Signal 2	2,72	2,32	3,94	1,70	407 ms
Signal 3	2,72	13,97	17,72	1,27	504 ms
Signal 4	2,87	4,13	5,24	1,27	564 ms
Signal 5	2,60	3,89	4,84	1,24	850 ms
Signal 6	2,60	4,21	5,18	1,23	706 ms
Signal 7	2,49	3,43	3,63	1,06	597 ms
Signal 8	2,62	3,38	3,86	1,14	518 ms

4.7 Application de la méthode rapide par fenêtre et débruitage par fenêtre en cumulant l'information progressivement

Dans cette section, notre objectif est d'implémenter la méthode rapide par fenêtre en appliquant le débruitage pour chaque fenêtre. Le calcul de A et B se fait en utilisant les fonctions d'autocorrélation cumulées jusqu'à la fenêtre courante. Les indicateurs de performance pour cette implémentation sont représentés dans les tableaux 4.12 et 4.13. Pour RI-1, on constate une faible amélioration de la qualité des signaux 8 et 1. Les autres signaux ont une valeur de SNRI entre 1,17 et 1,52, ce qui est une bonne amélioration. Ces valeurs sont comparables aux valeurs trouvées dans la section 4.3.2. Pour le signal de commandes agglomérées, la figure 4.17 montre que la valeur de SNRI est égale à 1,25 pour $L_c = 128$. De plus, les valeurs de PESQ, entre 2,22 et 2,71, montrent une bonne qualité perceptuelle des signaux. Puisque le traitement du signal se fait par fenêtre, le temps de calcul par fenêtre est égal au temps de calcul total divisé par le nombre de fenêtres. Le temps moyen de calcul pour une fenêtre de durée 32 ms ($L_f = 512$) est 41 ms, ce qui montre qu'on est très proche d'une solution en

temps réel.

Pour RI-4, avec $L_c = 128$, les valeurs de SNRI présentées dans le tableau 4.13 montrent une légère amélioration par rapport aux valeurs trouvées avec le signal de commandes dans la section 4.3.3 pour $L_c = 128$. En effet, la figure 4.23 montre que pour $L_c = 128$, la valeur de SNRI est 1,09 tandis que la médiane des SNRI dans le tableau 4.13 est de 1,19. Ici l'avantage principal est l'absence de SNRI inférieur à 1, alors qu'ils sont fréquents si on utilise un filtre fixe pour toutes les fenêtres, comme le montre la figure 4.25. On voit aussi que les valeurs de PESQ sont entre 2,27 et 2,77, ce qui est une bonne qualité perceptuelle des signaux. Les deux tableaux 4.12 et 4.13 montrent que lorsqu'on fait le débruitage par fenêtre, les temps de calcul sont plus longs que les temps de calcul obtenus si on fait le débruitage à la fin. Cette différence est due au temps de calcul de la matrice A ($L_c \times L_c$) et du vecteur B ($L_c \times 1$) qui devraient être calculés pour chaque fenêtre et pas seulement une fois à la fin. En comparant les temps de calcul obtenus et les durées des signaux présentées dans le tableau 4.3, on constate qu'on est proche du temps réel. Par exemple, le temps de calcul du signal 1 qui dure 1,36 s est 1.48 s. Les temps de calcul ont été obtenus en implémentant l'algorithme avec Python sur un ordinateur Acer Nitro 5 avec un processeur Intel 8 coeurs (2.21 GHz).

Tableau 4.12 Les mesures de performance de l'implémentation rapide par fenêtre pour les signaux de commande et RI-1

Signal	PESQ	SNR_i	SNR_f	SNRI	Temps de calcul
Signal 1	2,38	3,56	3,96	1,11	1,48 s
Signal 2	2,70	6,44	7,77	1,21	1,89 s
Signal 3	2,30	5,41	7,63	1,41	2,34 s
Signal 4	2,66	2,62	3,85	1,47	2,71 s
Signal 5	2,22	2,25	2,65	1,17	3,74 s
Signal 6	2,37	1,77	2,70	1,52	3,14 s
Signal 7	2,35	3,22	4,86	1,51	2,74 s
Signal 8	2,71	3,95	4,16	1,05	2,35 s
Moyenne des valeurs	2,46	3,65	4,70	1,31	2,55 s

Tableau 4.13 Les mesures de performance de l'implémentation rapide par fenêtre pour les signaux de commande et RI-4

Signal	PESQ	SNR_i	SNR_f	SNRI	Temps de calcul
Signal 1	2,57	2,01	3,36	1,67	1,87 s
Signal 2	2,57	2,32	4,19	1,80	2,02 s
Signal 3	2,27	13,97	16,23	1,16	2,45 s
Signal 4	2,77	4,13	4,45	1,08	2,84 s
Signal 5	2,47	3,89	4,63	1,19	3,93 s
Signal 6	2,53	4,21	4,83	1,15	3,45 s
Signal 7	2,39	3,43	3,81	1,11	2,67 s
Signal 8	2,72	3,38	4,04	1,19	2,41 s
Moyenne des valeurs	2,54	4,67	5,69	1,29	2,70 s

CHAPITRE 5 CONCLUSION ET RECOMMENDATIONS

5.1 Synthèse des travaux

L'objectif de ce mémoire était d'étudier une technique de déréverbération dans le but de réduire l'effet de la réverbération sur les signaux de parole et d'améliorer la performance du système de reconnaissance automatique de la parole de notre partenaire fluent.ai. On a commencé par une étude de la réverbération, ses caractéristiques et ses effets sur la parole. Ensuite, on a présenté différentes approches de déréverbération. Parmi, les techniques présentées, on a choisi la méthode de déréverbération par prédiction linéaire. La méthode considérée consiste à estimer un filtre qui sert à réduire la réverbération dans le signal observé par le microphone. Cette méthode nécessite beaucoup de calcul et le temps de traitement est élevé par rapport aux exigences de notre partenaire. Pour réduire le temps de calcul, on a proposé une méthode rapide qui se base sur le calcul de fonction d'autocorrélation par fenêtre. La fonction d'autocorrélation calculée dans une fenêtre est normalisée par la variance de la fenêtre courante. Les fonctions d'autocorrélation normalisées sont cumulées jusqu'à la fin du signal. Cette méthode est rapide car elle bénéficie de la transformée de Fourier rapide.
Pour évaluer les deux méthodes, on a utilisé des indicateurs de performance objectifs pour quantifier l'amélioration de la qualité du signal. Pour étudier le comportement de la méthode de déréverbération en fonction de la taille L_c du filtre et du décalage D, on a commencé par tester la méthode avec une source de bruit blanc. Ce test nous a permis d'avoir une idée sur le choix des valeurs des paramètres L_c et D. On a constaté pour deux différentes RI que la valeur de SNRI augmente si L_c augmente et se stabilise pour $L_c = L_h$. Pour $L_c = L_h$, on a obtenu SNRI $\approx$ 2,2 pour RI-1 et SNRI $\approx$ 4,5 pour RI-4, ce qui est une amélioration significative de la qualité du signal.
Avec une source de bruit blanc, on a étudié aussi l'effet de la partie d'apparence aléatoire de la RI sur les indicateurs de performance. On a trouvé que la performance de l'algorithme de déréverbération peut-être assez sensible à la structure de la partie irrégulière de la RI.Ces résultats montrent que la réverbération tardive, qui est très sensible à la position du locuteur, peut avoir un effet significatif sur la performance de la méthode.
Nous avons ensuite testé deux signaux de parole qu'on a supposé stationnaires. Le premier est un extrait d'entrevue qui dure 20 s. Le deuxième est une juxtaposition de huit signaux de commandes de durée 16 s. On a testé la méthode déréverbération avec ces deux signaux en calculant le filtre de déréverbération pour la totalité de chacun de deux signaux. L'objectif de ce test est de tester les performances globales et locales d'un filtre optimisé globalement.

Les SNRI globaux s'échelonnent entre 1,03 et 1,62 pour les deux signaux et les deux RI, donc la qualité du signal complet est améliorée. Pour le signal d'entrevue et RI-1, on a trouvé des SNRI locaux qui varient entre 1 et 2,5. Ces valeurs montrent qu'un filtre qui n'est pas adapté au signal localement peut aussi améliorer la qualité locale du signal. Cependant, pour les autres combinaisons signal-RI, nous avons constaté que les SNRI locaux pouvaient varier entre 0,4 et 4, ce qui montre les limitations d'un filtre optimisé globalement. Pour assurer une amélioration locale du signal, il est nécessaire d'optimiser le filtre localement. Ceci nous conduit au débruitage individuel des signaux de commande.

Nous avons débruité les signaux de commande individuellement avec la méthode de base pour ces signaux avec $L_c = 500$. Pour RI-1, on a obtenu une valeur moyenne de SNRI=1,65 pour tous les signaux. Cette valeur montre que le calcul du filtre localement avec la méthode de base est plus performant que le calcul du filtre globalement. L'avantage principal de l'optimisation locale du filtre est que la qualité du signal traité est toujours égale ou supérieure à celle du signal initial. De plus, les valeurs de PESQ trouvées, entre 2 et 3, montrent que les signaux estimés ont une bonne qualité perceptuelle. Cependant, on a trouvé que le temps de calcul par la méthode de base est loin de ce qu'on veut avoir pour l'application visée.

Dans la pratique, le traitement de la parole se fait par fenêtre. Pour cette raison, on a implémenté la méthode de base par fenêtre en appliquant le débruitage à la fin. Dans cette implémentation, on a supposé que la variance était constante sur des fenêtres de taille $L_f = 512$ et on fixé $L_c = 128$. On a constaté que le temps de calcul était environ cinq fois plus petit. Certes, la valeur du SNRI est plus petite que celle trouvée avec la méthode de base, donc sans fenêtre. La réduction du SNRI et du temps de calcul s'explique par la réduction de la taille L_c du filtre. Les valeurs de PESQ sont comparables pour les deux implémentations de la méthode de base. Comme les temps de calcul obtenus avec l'implémentation de la méthode de base par fenêtre sont encore trop grands par rapport aux attentes de notre partenaire, on a ensuite testé l'implémentation de la méthode rapide par fenêtre avec débruitage à la fin.

L'implémentation rapide a permis de réduire encore le temps de calcul d'un facteur 38, ce qui devient significatif pour l'application de la déréverbération en temps réel. Les valeurs de SNRI obtenus, qui varient entre 1,06 et 2,14, correspondent à une amélioration. Les valeurs du PESQ correspondent aussi à une bonne qualité perceptuelle.

Notre partenaire souhaite avoir une solution implémentée en temps réel. Cela implique que le traitement doit être fait sur chaque fenêtre dès qu'une nouvelle fenêtre de données devient disponible. Ceci nous a amené à tester l'implémentation de la méthode rapide par fenêtre en effectuant le débruitage fenêtre par fenêtre, et non à la fin, et en cumulant l'information progressivement. Pour ce test, la valeur moyenne du SNRI est 1,31 pour RI-1 et 1,29 pour RI-4 ce qui correspond à une amélioration de la qualité des signaux de commandes. Les valeurs du

PESQ obtenues correspondent aussi à une bonne qualité perceptuelle des signaux débruités. De plus, on a trouvé que le temps moyen de calcul pour une fenêtre de durée 32 ms (512 points) est de 41 ms, ce qui montre qu'on est très proche d'une solution en temps réel.

5.2 Limitations de la solution proposée

La méthode rapide par fenêtre permet d'améliorer le signal en temps quasi-réel. Cependant, nous ne savons pas si l'amélioration du signal est suffisante pour améliorer la performance de l'engin de reconnaissance de la parole de notre partenaire.
Remarquons que l'utilisation de fenêtre de 512 points nous force à utiliser une longueur de filtre assez petite ($L_c = 128$), ce qui limite la performance du filtre. En effet, nous avons montré que sa performance augmente significativement dans l'intervalle $128 \leq L_c \leq 800$.

5.3 Améliorations futures

Une implémentation de la méthode de base dans le domaine fréquentiel pourrait peut-être améliorer sa performance [33]. L'utilisation de données provenant de plusieurs microphones serait aussi une avenue pour avoir plus d'information sur la réverbération et pour obtenir un filtre de déréverbération plus performant [34]. Il serait aussi intéressant de développer une méthode pour tester quantitativement l'effet du débruitage sur l'engin de reconnaissance de la parole de notre partenaire.

RÉFÉRENCES

[1] T. Nakatani *et al.*, "Speech dereverberation based on variance-normalized delayed Linear prediction," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 18, n°. 7, p. 1717–1731, Spetembre 2010. [En ligne]. Disponible : http://ieeexplore.ieee.org/document/5547558/

[2] E. Habets, "Single- and multi-microphone speech dereverberation using spectral enhancement," thèse de doctorat, Department of Electrical Engineering, Technische Universiteit Eindhoven, Eindhoven, 2007. [En ligne]. Disponible : https://research.tue.nl/en/publications/single-and-multi-microphone-speech-dereverberation-using-spectral

[3] T. Yoshioka *et al.*, "Making machines understand us in reverberant rooms : Robustness against reverberation for automatic speech recognition," *IEEE Signal Processing Magazine*, vol. 29, n°. 6, p. 114–126, Octobre 2012. [En ligne]. Disponible : https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=6296524

[4] H. Dudley et T. H. Tarnoczy, "The speaking machine of wolfgang von kempelen," *The Journal of the Acoustical Society of America*, vol. 22, n°. 2, p. 151–166, 1950.

[5] J. Benesty, M. M. Sondhi et Y. Huang, *Springer handbook of speech processing*. Springer, 2007.

[6] K. H. Davis, R. Biddulph et S. Balashek, "Automatic recognition of spoken digits," *The Journal of the Acoustical Society of America*, vol. 24, n°. 6, p. 637–642, 1952.

[7] R. Gemello et F. Mana, "Noise reduction for automatic speech recognition," Brevet, 20 may 2008, uS Patent 7,376,558. [En ligne]. Disponible : https://patentimages.storage.googleapis.com/e9/62/70/b4f8f6f90a257c/US7376558.pdf

[8] N. Patrick et N. D. Gaubitch, "Introduction," dans *Speech Dereverberation*, N. D. G. Patrick Naylor, édit. London, UK : Springer-Verlag London, 2010, p. 17–35. [En ligne]. Disponible : https://www.springer.com/gp/book/9781849960557

[9] P. C. Loizou, "Speech quality assessment," dans *Multimedia Analysis, Processing and Communications*, W. Lin *et al.*, édit. Berlin, Allemagne : Springer-Verlag Berlin Heidelberg, 2011, p. 623–654. [En ligne]. Disponible : https://link.springer.com/book/10.1007/978-3-642-19551-8

[10] W. Steven, "Psychoacoustic influences of the echoing environments of prehistoric art," *Journal of The Acoustical Society of America*, vol. 112, p. 2284–2284, Novembre 2002. [En ligne]. Disponible : https://www.researchgate.net/publication/253473133_Psychoacoustic_influences_of_the_echoing_environments_of_prehistoric_art

[11] R. H. Bolt et A. D. MacDonald, “Theory of speech masking by reverberation,” *The journal of the acoustical society of America*, vol. 21, n°. 6, p. 577–580, 1949. [En ligne]. Disponible : https://asa.scitation.org/doi/10.1121/1.1906551

[12] H. Haas, “The influence of a single echo on the audibility of speech,” *Journal of the audio engineering society*, vol. 20, n°. 2, p. 146–159, Mars 1972. [En ligne]. Disponible : https://www.aes.org/e-lib/online/browse.cfm?elib=2093

[13] N. López, “Méthodes parcimonieuses pour la déreverbération des signaux audio,” mémoire de maîtrise, Département de communication, Université TELECOM ParisTech, Paris, France, 2011. [En ligne]. Disponible : ftp://ftp.ircam.fr/pub/IRCAM/equipes/repmus/Atiam/Lopez.pdf

[14] H. Kuttruff, *Room acoustics*. Crc Press, 2016.

[15] W. C. Sabine, *Collected papers on acoustics*. USA : Cambridge : Harvard University Press, 1922.

[16] Y. Hu et K. Kokkinakis, “Effects of early and late reflections on intelligibility of reverberated speech by cochlear implant listeners,” *The Journal of the Acoustical Society of America*, vol. 135, n°. 1, p. EL22–EL28, Janvier 2014. [En ligne]. Disponible : https://asa.scitation.org/doi/pdf/10.1121/1.4834455

[17] A. Belhomme, “Méthodes de déréverbération tardive de la parole,” mémoire de maîtrise, Département de communication, Université TELECOM ParisTech, Paris, France, 2014. [En ligne]. Disponible : http://repmus.ircam.fr/_media/atiam/Memoire_BELHOMME.pdf

[18] S. Gannot, D. Burshtein et E. Weinstein, “Signal enhancement using beamforming and nonstationarity with applications to speech,” *IEEE Transactions on Signal Processing*, vol. 49, n°. 8, p. 1614–1626, Aout 2001. [En ligne]. Disponible : https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=934132

[19] S. Affes et Y. Grenier, “A signal subspace tracking algorithm for microphone array processing of speech,” *IEEE Transactions on Speech and Audio Processing*, vol. 5, n°. 5, p. 425–437, Septembre 1997.

[20] M. Brandstein, *Microphone arrays : signal processing techniques and applications*. Springer Science & Business Media, 2001.

[21] A. Plinge *et al.*, “Acoustic microphone geometry calibration : An overview and experimental evaluation of state-of-the-art algorithms,” *IEEE Signal Processing Magazine*, vol. 33, n°. 4, p. 14–29, Juillet 2016.

[22] K. Lebart, J.-M. Boucher et P. N. Denbigh, "A new method based on spectral subtraction for speech dereverberation," *Acta Acustica united with Acustica*, vol. 87, n°. 3, p. 359–366, 2001.

[23] E. A. Habets, "Multi-channel speech dereverberation based on a statistical model of late reverberation," dans *Proceedings.(ICASSP'05). IEEE International Conference on Acoustics, Speech, and Signal Processing, 2005.*, vol. 4. IEEE, 2005, p. iv–173.

[24] J.-D. Polack, "La transmission de l'énergie sonore dans les salles," thèse de doctorat, Le Mans, 1988.

[25] T. Yoshioka *et al.*, "Statistical models for speech dereverberation," dans *2009 IEEE Workshop on Applications of Signal Processing to Audio and Acoustics*. IEEE, 2009, p. 145–148.

[26] T. Hikichi, M. Delcroix et M. Miyoshi, "Inverse filtering for speech dereverberation less sensitive to noise and room transfer function fluctuations," *EURASIP Journal on Advances in Signal Processing*.

[27] M. Miyoshi et Y. Kaneda, "Inverse filtering of room acoustics," *IEEE Transactions on acoustics, speech, and signal processing*, vol. 36, n°. 2, p. 145–152, 1988.

[28] S. T. Neely et J. B. Allen, "Invertibility of a room impulse response," *The Journal of the Acoustical Society of America*, vol. 66, n°. 1, p. 165–169, 1979.

[29] T. Dietzen *et al.*, "Joint multi-microphone speech dereverberation and noise reduction using integrated sidelobe cancellation and linear prediction," dans *2018 16th International Workshop on Acoustic Signal Enhancement (IWAENC)*. IEEE, 2018, p. 221–225.

[30] M. Parchami, W.-P. Zhu et B. Champagne, "Speech dereverberation using weighted prediction error with correlated inter-frame speech components," *Speech communication*, vol. 87, p. 49–57, 2017.

[31] Y. Benabderrahmane, "Séparation aveugle de signaux de parole utilisant les statistiques d'ordre supérieur et la décomposition en sous-espaces," thèse de doctorat, Centre Énergie Matériaux Télécommunications, Institut national de la recherche scientifique, Québec, QC, 2011. [En ligne]. Disponible : http://espace.inrs.ca/id/eprint/2132/

[32] *Evaluation de la qualité vocale perçue : méthode objective d'évaluation de la qualité vocale de bout en bout des codecs vocaux et des réseaux téléphoniques à bande étroite*, Norme UIT-T P.862, 2001.

[33] T. Nakatani *et al.*, "Speech dereverberation in short time fourier transform domain with crossband effect compensation," dans *2008 Hands-Free Speech Communication and Microphone Arrays*. IEEE, 2008, p. 220–223.

[34] ——, “Blind speech dereverberation with multi-channel linear prediction based on short time fourier transform representation,” dans *2008 IEEE International Conference on Acoustics, Speech and Signal Processing*. IEEE, 2008, p. 85–88.

ANNEXE A LES INDICATEURS DE LA QUALITÉ DE DÉBRUITAGE POUR UNE SOURCE DE BRUIT BLANC

Calcul analytique des indicateurs de la qualité de débruitage pour une source de bruit blanc

La définition des indicateurs de performance a été donnée à la section 4.1.1. Leurs expressions dépendent de $\mathbb{E}[(d(n))^2]$, $\mathbb{E}[(r(n))^2]$ et $\mathbb{E}[(\hat{r}(n) - r(n))^2]$, dont nous dérivons les expressions de ces quantités dans cette section.

Calcul de $\mathbb{E}[(d(n))^2]$ et $\mathbb{E}[(r(n))^2]$

Comme $d = \sum_{k=0}^{D-1} h_k \; s(n-k)$ et $r = \sum_{k=D}^{L_h-1} h_k \; s(n-k)$, l'indépendance des $s(n)$ implique que

$$\mathbb{E}[(d(n))^2] = \sigma^2 \sum_{k=0}^{D-1} (h_k)^2 = \sigma^2 ||h_1||^2,$$

et

$$\mathbb{E}[(r(n))^2] = \sigma^2 \sum_{k=D}^{L_h-1} (h_k)^2 = \sigma^2 ||h_2||^2,$$

où $||h_1||^2 := \sum_{k=0}^{D-1} (h_k)^2$ et $||h_2||^2 := \sum_{k=D}^{L_h-1} (h_k)^2$.

Calcul de $\mathbb{E}[(\hat{r}(n) - r(n))^2]$

Tout d'abord, notons que

$$\mathbb{E}[(\hat{r}(n) - r(n))^2] = \mathbb{E}[(\hat{r}(n))^2] + \mathbb{E}[(r(n))^2] - 2\mathbb{E}[r(n)\hat{r}(n)].$$

Comme $\hat{r}(n) = \hat{C}^T X(n-D)$, on a

$$\mathbb{E}[(\hat{r}(n))^2] = \mathbb{E}[\hat{C}^T X(n-D) X^T(n-D) \hat{C}] = \hat{C}^T A \hat{C},$$

où $A := \mathbb{E}[X(n-d)X^T(n-D)]$. Le vecteur $\hat{C}^T$ est la solution du problème d'optimisation

$$\min_C F(C),$$

où

$$\begin{aligned} F(C) &:= \mathbb{E}\left[(x(n) - C^T X(n-D))^2\right], \\ &= \mathbb{E}\left[(x(n))^2 + C^T X(n-D) X^T(n-D) C - 2C^T x(n) X(n-D)\right], \\ &= \sigma^2 + C^T A\, C - 2\, C^T B, \end{aligned} \tag{A.1}$$

où $B := \mathbb{E}[x(n)X(n-D)]$. On a supposé ici que la variance du signal désiré est constante. La condition d'optimalité d'ordre 1 pour F est $\nabla_C F|_{C=\hat{C}} = 0$, qui conduit à

$$A\hat{C} = B. \tag{A.2}$$

Les matrices A et B satisfont

$$A_{i,j} = \mathbb{E}[x(n-D-i)x(n-D-j)] = R_x(|i-j|), \tag{A.3}$$

$$B_i := \mathbb{E}[x(n)x(n-D-i)] = R_x(D+i) \tag{A.4}$$

où $R_x(\ell) := \mathbb{E}[x(n)x(n+\ell)]$. Si la source est un bruit blanc stationnaire de moyenne nulle et de variance σ^2, l'expression de la fonction R_x peut être obtenue comme suit :

$$\begin{aligned} R_x(\ell) &= \mathbb{E}\left[\left(\sum_{i=0}^{L_h-1} h_i\ s(n-i)\right)\left(\sum_{j=0}^{L_h-1} h_j\ s(n+\ell-j)\right),\right] \\ &= \sum_{i=0}^{L_h-1}\sum_{j=0}^{L_h-1} h_i\ h_j\ R_s(\ell+i-j), \\ &= \sum_{i=0}^{L_h-1} h_i \sum_{j=0}^{L_h-1} h_j\ \sigma^2\, \delta_{j,\,i+\ell}, \\ &= \sigma^2 \sum_{i=0}^{L_h-1} h_i\ h_{i+\ell}, \\ &= \sigma^2 \sum_{i=0}^{L_h-1-\ell} h_i\ h_{i+\ell} \text{ car } h_i = 0 \text{ si } i > L_h - 1 \text{ ou } i < 0, \\ &= \sigma^2\ R_h(\ell), \end{aligned} \tag{A.5}$$

où

$$R_h(\ell) := \sum_{j=0}^{L_h-1-\ell} h_{j+\ell}\ h_j.$$

On a donc

$$A_{i,j} = \sigma^2\ R_h(i-j),$$

$$B_i = \sigma^2\ R_h(D+i).$$

L'expression de $\mathbb{E}\left[r(n)\,\hat{r}(n)\right]$ peut être obtenue comme suit :

$$\begin{aligned}
\mathbb{E}\left[r(n)\,\hat{r}(n)\right] &= \mathbb{E}\left[\left(\sum_{k=D}^{L_h-1} h_k\ s(n-k)\right)\left(\sum_{j=0}^{L_c-1} \hat{C}_j\ x(n-D-j)\right)\right] \\
&= \mathbb{E}\left[\sum_{k=D}^{L_h-1}\sum_{j=0}^{L_c-1} h_k\ s(n-k)\ \hat{C}_j \sum_{i=0}^{L_h-1} h_i\ s(n-D-j-i)\right], \\
&= \sigma^2 \sum_{j=0}^{L_c-1} h_{D+i+j}\ \hat{C}_j \sum_{i=0}^{L_h-1} h_i \text{ si } s \text{ est un bruit blanc}, \\
&= \sigma^2 \sum_{j=0}^{L_c-1} \hat{C}_j \sum_{i=0}^{L_h-1} h_i\ h_{D+i+j}, \\
&= \sigma^2 \sum_{k=D}^{L_c-1+D} \hat{C}_{k-D} \sum_{i=0}^{L_h-1} h_i\ h_{i+k}, \\
&= \sigma^2 \sum_{k=D}^{L_c-1+D} \hat{C}_{k-D} \sum_{i=0}^{L_h-1-k} h_i\ h_{i+k} \text{ car } h_i = 0 \text{ si } i > L_h-1, \\
&= \sigma^2 \sum_{k=D}^{L_c-1+D} \hat{C}_{k-D}\ R_h(k), \\
&= \sigma^2 \sum_{k=0}^{L_c-1} \hat{C}_k\ R_h(D+k).
\end{aligned} \tag{A.6}$$

On a donc

$$\mathbb{E}\left[(\hat{r}(n)-r(n))^2\right] = \hat{C}^T A\,\hat{C} + \sigma^2\,||h_2||^2 - 2\,\sigma^2 \sum_{k=0}^{L_c-1} \hat{C}_k\ R_h(D+k).$$