國 立 清 華 大 學

碩 士 論 文

聲學傳遞函數盲估計以應用於去混響、聲源分離以及增強

Blind estimation of acoustic transfer functions with application to signal dereverberation, source separation, and speech enhancement

系級：動力機械工程學系碩士班

組別：電機控制組

學號姓名：111033537 袁安志 Anchi Yuan

指導教授：白明憲 博士 (Dr. Mingsian R. Bai)

中 華 民 國 一 一 三 年 七 月

# 摘要

雖然在陣列信號處理中，聲學傳遞函數（Acoustic Transfer Functions）通常比相對傳遞函數（Relative Transfer Functions）具有更好的性能，但由於源輸入信號通常不可用，獲得可靠的聲學傳遞函數估計具有挑戰性。為了解決這一問題，我們提出了一種基於卷積傳遞函數（Convolutive Transfer Functions）的創新盲聲學傳遞函數估計方法。我們首先使用到達時間差（Time Difference of Arrival）和廣義互相關相位變換（Generalized Cross Correlation-Phase Transform）估計來定位分佈式陣列中的聲源。接著，我們應用加權預測誤差（Weighted Prediction Error）算法對混合緊湊-分佈式陣列接收到的信號進行去混響，並使用延遲和求和波束形成器作為源信號的初步估計。卷積傳遞函數係數可以使用維納濾波器或卡爾曼濾波器計算，並使用粒子群優化（Particle Swarm Optimization）優化其參數。數值模擬和使用十三麥克風混合陣列進行的實驗證明了所提出技術的有效性。最先進的自適應多通道時域最小均方（Adaptive Multi-channel Time Domain Least Mean Square）方法被用作基線。為了進一步驗證，我們將所提出的方法應用於信號去混響、聲源分離和語音增強等應用。

***關鍵詞 ― 卷積傳遞函數，加權預測誤差算法，延遲和加總波束成形器，維納濾波器，卡爾曼濾波器，粒子群優化***

# ABSTRACT

While Acoustic Transfer Functions (ATFs) generally lead to better performance than Relative Transfer Functions (RTFs) in array signal processing, obtaining reliable ATF estimates is challenging because the source input is usually unavailable. To address this problem, we propose a novel blind ATF estimation approach formulated using Convolutive Transfer Functions (CTFs). We start by locating the source using Time Difference of Arrival (TDOA) estimated by Generalized Cross Correlation-Phase Transform (GCC-PHAT), by using a distributed array. Next, we apply the Weighted Prediction Error (WPE) algorithm to de-reverberate the signals received by a hybrid compact-distributed array, using the Delay and Sum beamformer as an initial estimate of the source signal. The CTF coefficients can be computed using either the Wiener filter or the Kalman filter with the parameters optimized using Particle Swarm Optimization (PSO). Simulations and experiments using a thirteen-microphone hybrid array demonstrate the efficacy of the proposed technique. The state-of-the-art Adaptive Multichannel Time Domain Least Mean Square (MCLMS) method was used as the baseline. For further validation, we applied the proposed technique to applications, including signal dereverberation, source separation, and speech enhancement.

***Index Terms — convolutive transfer functions, weighted prediction error, delay and sum beamformer, Wiener filter, Kalman filter, particle swarm optimization***

# 致謝

時光飛逝歲月如梭，轉眼間碩士生涯也即將邁入尾聲，首先感謝指導教授—白明憲教授的諄諄教誨，在研究上給予許多靈感與導正，如今也才能順利進行口試，這些日子也在老師的身上看到了對於研究的不放棄與熱忱，這也勉勵我未來不論是對於工作甚至是生活都應該抱持相同的態度，且實驗室的生活也讓我了解到如何將理論應用於實際情境下，這些累積的經驗對於我來說可謂是收穫滿滿。

非常感謝教授及教授願意於百忙之中抽空擔任口試委員，且撥冗閱讀並指導學生的口試論文並使其更加完整，於此獻上無限的感激。

求學生涯中特別感謝家人們尤其是父母在我遇到困難與挫折時給予各方面上的支持與鼓勵，也感謝身邊的朋友不離不棄的陪伴，最後感謝同實驗室的陳佑祥學長、許逸誠學長、孔繁傑學長、賴柏儒學長、林邑軒學長、吳仲倫學長、張馨予學姐及陳思涵學姊在學業上的幫助與指導，得以讓我及時解決研究上的問題，且有幸與同學曾大容、鍾沛霖、陳星宇、李鼎珅、于芷萱、范家萱及李可欣互相討論學業，一起成長，承蒙一路上支持與幫助我的貴人，由衷感謝大家。

# CONTENTS

[摘要 ii](#_Toc170759040)

[ABSTRACT iii](#_Toc170759041)

[致謝 v](#_Toc170759042)

[CONTENTS vi](#_Toc170759043)

[LIST OF ALGORITHMS viii](#_Toc170759044)

[LIST OF FIGURES ix](#_Toc170759045)

[LIST OF TABLES x](#_Toc170759046)

[Chapter 1. INTRODUCTION 1](#_Toc170759047)

[Chapter 2. BASELINE APPROACH 5](#_Toc170759048)

[Chapter 3. CTF SIGNAL MODEL 11](#_Toc170759049)

[3.1. Representation of LTI Systems in Crossband Filter 11](#_Toc170759050)

[3.2. Band-to-band Filter as CTF Signal Model 14](#_Toc170759051)

[Chapter 4. PROPOSED METHOD 18](#_Toc170759052)

[4.1. TDOA-based Source Localization 18](#_Toc170759053)

[4.1.1. GCC-PHAT for TDOA estimation 19](#_Toc170759054)

[4.1.2. TDOA Measurement Model 20](#_Toc170759055)

[4.1.3. Constrained Least Squares (CLS) method 23](#_Toc170759056)

[4.2. Pre-processing 24](#_Toc170759057)

[4.2.1. WPE 24](#_Toc170759058)

[4.2.2. DAS Beamformer 28](#_Toc170759059)

[4.3. Wiener Filtering Approach 29](#_Toc170759060)

[4.4. RLS Approach 30](#_Toc170759061)

[4.5. Kalman Algorithm Adaptive Filtering Approach 31](#_Toc170759062)

[4.6. ATF Reconstruction 33](#_Toc170759063)

[4.7. Parameters Optimization 34](#_Toc170759064)

[4.7.1. PSO 35](#_Toc170759065)

[4.7.2. ASPSO 36](#_Toc170759066)

[4.8. Summary of Proposed Method 41](#_Toc170759067)

[Chapter 5. SIMULATIONS 43](#_Toc170759068)

[5.1. Fixed Source Location Cases 43](#_Toc170759069)

[5.2. Fixed Source Location with Parameters Optimization 46](#_Toc170759070)

[5.3. Applications 46](#_Toc170759071)

[5.4. Moving Source Location Cases 46](#_Toc170759072)

[Chapter 6. EXPERIMENTS 47](#_Toc170759073)

[6.1. Experiment Settings and Parameters 47](#_Toc170759074)

[6.2. Experiment Results and Discussions 47](#_Toc170759075)

[Chapter 7. CONCLUSIONS AND FUTURE WORK 48](#_Toc170759076)

[7.1. Conclusions 48](#_Toc170759077)

[7.2. Future Work 48](#_Toc170759078)

[REFERENCES 49](#_Toc170759079)

# LIST OF ALGORITHMS

[Algorithm 1 WPE 27](#_Toc170759080)

[Algorithm 2CTF estimation using Wiener filtering 30](#_Toc170759081)

[Algorithm 3 CTF estimation using RLS 31](#_Toc170759082)

[Algorithm 4 CTF estimation using stationary Kalman adaptive filtering 32](#_Toc170759083)

# LIST OF FIGURES

[Figure 1 The sound sources propagation model 20](#_Toc170759084)

[Figure 2 Block diagram of PSO 36](#_Toc170759085)

[Figure 3 Block diagram of ASPSO 41](#_Toc170759086)

[Figure 4 Configuration of the room for fixed source location 46](#_Toc170759087)

# LIST OF TABLES

[Table 1 Flow chart of our proposed method 41](#_Toc170759088)

[Table 2 Specifications of room settings for different RIRs 44](#_Toc170759089)

# INTRODUCTION

Blind estimation, namely blind system identification (BSI), pertains to identifying systems where solely the output signals are known, and a minimal amount of data is known regarding the input signals. This challenge is highly significant due to the practical applications of estimated Acoustic Transfer Functions (ATFs) in various acoustic scenarios such as acoustic echo cancellation [1] , dereverberation [2] , blind source separation [3] , and beamforming in reverberant environments [4] . Most BSI techniques have typically functioned using the time domain [5] or the Short Time Fourier Transform (STFT) domain [6] [7] , where they estimate convolution in the time domain by multiplying the source STFT with the room impulse response (RIR) STFT. This approximation, called the multiplicative transfer function (MTF) approximation [8] or the narrowband approximation, is valid in theory only if the RIR's length is shorter than that of the STFT window. However, in practical scenarios, this requirement is rarely met, even in moderately reverberant environments. The limitations of the STFT window in assuming local stationarity of audio signals result in this. Additionally, the use of a long STFT window can lead to increased estimation variance and computational complexity.

To tackle this problem, especially in situations involving extended RIRs, Crossband Filters (CBFs) were introduced in [9] for linear system identification. These CBFs provide an alternative to the MTF approach. In this alternative, the STFT coefficient output is represented as the sum of multiple convolutions between the STFT coefficients of the input source signal and the RIR in the time-frequency (TF) domain along the frame coordinate. For analytical tractability, an approximation of CBFs called the convolutive transfer function (CTF) [10] has been proposed. This model proposes that, for each frequency, the STFT coefficient output can be represented as a distinct convolution between the STFT coefficients of the input source signal and the CTF along the frame axis.

This thesis outlines a methodology for the estimation of blind ATF through CTF approximation. To start the process, source localization algorithm is needed to obtain accurate source location in the space for the source signal extraction of the next stage. We, in this thesis, use Generalized Cross Correlation-Phase Transform (GCC-PHAT) [11] to estimate Time Difference of Arrival (TDOA) [12] of each microphone in a distributed array. Once source location was acquired, we can move to dereverberation and source signal extraction using Weighted Prediction Error (WPE) [13] and Delay and Sum (DAS) beamforming [14] . Afterward, CTF coefficients are computed using Wiener filters [15] or adaptive filters, such as Recursive Least Squares (RLS) [16] and Kalman filter [17] . The parameters of previous mentioned filters are optimized using Particle Swarm Optimization (PSO) [18] and its enhanced version [19] . To acquire ATFs in the time domain, namely Room Impulse Responses (RIRs), from the CTF coefficients, the estimated CTF coefficients are convolved with the constant magnitude filter across frequency axis. Subsequently, the resulting convolved sequence is subjected to processing using the inverse STFT.

The convergence performance is assessed using the normalized root mean square projection misalignment (NRMSPM) between the ground truth RIR and the estimated RIR. The results are compared with the baseline approach, specifically the Adaptive Multichannel Time Domain Least Mean Square (MCLMS) [20] method. Simulations encompass a wide range of reverberation times from 0.01 seconds to 1.6 seconds, utilizing a hybrid compact-distributed array of 38 microphones. Applications include signal dereverberation using the Multiple Input/Output Inverse Theorem (MINT) [21] , source separation via Tikhonov Regularization (TIKR) [22] , and speech enhancement using the Minimum Power Distortionless Response (MPDR) [23] beamformer, all of which require precise ATF estimates. The effectiveness of these applications is evaluated using several metrics, including the Perceptual Evaluation of Speech Quality (PESQ) [24] and Signal-to-Distortion Ratio (SDR) [25] . Additionally, experiments are conducted in a realistic room with a reverberation time of 0.128 seconds, using a hybrid compact-distributed array of 13 microphones. The results from both simulations and experiments demonstrate that the proposed approach outperforms the MCLMS method in various reverberant environments.

# BASELINE APPROACH

In this section, we introduce a state-of-the-art blind system identification algorithm known as Adaptive Multichannel Time Domain Least Mean Square (MCLMS), which we adopt as a baseline approach to compare with our proposed ATF estimation method. This approach constructs an error signal based on the cross-relations between different channels in a novel and systematic manner, as detailed below.

The *i*-th observation *xi*(*n*) is the result of a linear convolution between the source signal *s*(*n*) and the corresponding channel response *hi*. This relationship is expressed as follows:

|  |  |
| --- | --- |
| , | ( ) |

where the \* indicates linear convolution with respect to the time index *n* and *M* is the number of channels. In vector form, the relationship between the input and the observation for the *i*-th channel is written as:

|  |  |
| --- | --- |
| , | ( ) |

where

|  |  |
| --- | --- |
| , | ( ) |

and *T* denotes matrix transpose. The channel parameter matrix **H***i* is of dimension *L* × (2*L-*1) and is constructed from the channel’s impulse response:

|  |  |
| --- | --- |
| , | ( ) |

where *L* is set to the length of the longest channel impulse response by assumption.

When the input signal is unknown, the cross-relation between sensor outputs can be utilized to estimate the channel impulse responses. This is based on the premise that

|  |  |
| --- | --- |
| . | ( ) |

However, during prediction, this cross-relation does not hold, leading to the definition of an error signal as follows:

|  |  |
| --- | --- |
| . | ( ) |

We have (*M-*1)*M*/2distinct error signals *eij*(*n*) , excluding the case where *eii*(*n*) *=* 0and counting the pair *eij*(*n*) = -*eji*(*n*) only once. Assuming the equal importance of these error signals, we define a cost function as:

|  |  |
| --- | --- |
| . | ( ) |

The channel impulse responses are then defined by minimizing this error function. To avoid a trivial all-zero estimate, a unit-norm constraint is imposed on **h =** [**h**1T **h**2T…**h**MT]T at all times, such that the error signal becomes:

|  |  |
| --- | --- |
| . | ( ) |

The corresponding cost function is given by:

|  |  |
| --- | --- |
| . | ( ) |

The desired solution for **h** is determined by minimizing the mean value of the cost function *J*(*n*):

|  |  |
| --- | --- |
| . | ( ) |

Direct minimization is a computationally intensive process that may be intractable for long channel impulse responses and a large number of channels. Consequently, an LMS algorithm is proposed as a solution to this minimization problem, offering an efficient solution:

|  |  |
| --- | --- |
| , | ( ) |

where μ represents a small positive step size, while ∇ denotes the gradient operator. In order to ascertain the gradient in ( 11 ) , it is necessary to take the derivative of *J*(*n*) with respect to **h**:

|  |  |
| --- | --- |
| , | ( ) |

where

|  |  |
| --- | --- |
| . | ( ) |

We will now evaluate the partial derivative of χ(*n*) with respect to the coefficients of the *k*-th (*k* = 1,2, …, *M*) channel impulse response:

|  |  |
| --- | --- |
| , | ( ) |

where the final step follows from the fact that *ekk*(*n*) = 0. This equation may be expressed in matrix form concisely as follows:

|  |  |
| --- | --- |
| , | ( ) |

where we have defined, for the sake of convenience,

|  |  |
| --- | --- |
| . | ( ) |

Subsequently, the two matrix products in ( 15 ) are evaluated individually as follows:

|  |  |
| --- | --- |
| , | ( ) |

where

|  |  |
| --- | --- |
| . | ( ) |

Substitution of ( 17 ) into ( 15 ) yields the following result:

|  |  |
| --- | --- |
| . | ( ) |

Subsequently, we integrate ( 19 ) into ( 12 ) and derive the following result:

|  |  |
| --- | --- |
| , | ( ) |

where

|  |  |
| --- | --- |
| . | ( ) |

Finally, we substitute ( 20 ) into ( 11 ) and obtain the updated equation:

|  |  |
| --- | --- |
| . | ( ) |

If the channel estimate is always normalized following each update, then the simplified algorithm can be implemented as follows:

|  |  |
| --- | --- |
| . | ( ) |

Assuming that the independence assumption [26] holds, it can be demonstrated that the LMS algorithm converges in the mean if the step size satisfies

|  |  |
| --- | --- |
| , | ( ) |

where the largest eigenvalue of the matrix *E*{} is denoted by λmax.

# CTF SIGNAL MODEL

3.1. Representation of LTI Systems in Crossband Filter

In this section, we provide a concise overview of how digital signals and LTI systems are represented in the STFT domain. For more extensive information, please refer to sources such as [27] and [28] . First, we establish links between the crossband filters in the STFT domain and the impulse response in the time domain using analysis and synthesis windows. Unless stated otherwise, our summation indexes range from -∞ to ∞.

The STFT representation of a signal *x*(*n*) is given by

|  |  |
| --- | --- |
| , | ( 25 ) |

where

|  |  |
| --- | --- |
| . | ( 26 ) |

An analysis window of length *N* is denoted by [*w̃*](https://zh.wiktionary.org/zh-hant/Appendix:%E5%9B%BD%E9%99%85%E9%9F%B3%E6%A0%87%E7%AC%A6%E5%8F%B7#w%CC%83)(*n*). The frame index is denoted by *p* ∈ [1, *P*], and *k* ∈ [0, *N–*1] represents the frequency-band index. The discrete-time shift is denoted by *Ls*. Complex conjugation is represented by \*. The reconstruction of *x*(*n*) which is inverse STFT is achieved by

|  |  |
| --- | --- |
| , | ( 27 ) |

where

|  |  |
| --- | --- |
| , | ( 28 ) |

and *w*(*n*) denotes a synthesis window of length *N*. This thesis assumes [*w̃*](https://zh.wiktionary.org/zh-hant/Appendix:%E5%9B%BD%E9%99%85%E9%9F%B3%E6%A0%87%E7%AC%A6%E5%8F%B7#w%CC%83)(*n*) and *w*(*n*) are real functions. By substituting ( 25 ) into ( 27 ), we acquire the completeness condition

|  |  |
| --- | --- |
| . | ( 29 ) |

If the analysis and synthesis windows meet the requirements outlined in ( 29 ), the signal *x*(*n*) can be reconstructed flawlessly using its STFT coefficients *xp,k*. However, for *Ls*≦*N* and for a given synthesis window *w*(*n*), there might be an infinite number of solutions to ( 29 ); thus, the choice of the analysis window may not be unique according to [29] and [30] .

We will now delve into the STFT representation of LTI systems. Let *h*(*n*) denote the impulse response of an LTI system with a length of *Q*, where the input *x*(*n*) and output *o*(*n*) of this system are connected through the relation as follow:

|  |  |
| --- | --- |
| . | ( 30 ) |

From ( 25 ) and ( 30 ), the STFT of *o*(*n*) can be written as

|  |  |
| --- | --- |
| . | ( 31 ) |

Substituting ( 27 ) into ( 31 ), we obtain

|  |  |
| --- | --- |
| , | ( 32 ) |

where

|  |  |
| --- | --- |
|  | ( 33 ) |

may be interpreted as the STFT of *h*(*n*) using a composite analysis window ∑*m wpˊ,k*ˊ(*m-l*) [*w̃*](https://zh.wiktionary.org/zh-hant/Appendix:%E5%9B%BD%E9%99%85%E9%9F%B3%E6%A0%87%E7%AC%A6%E5%8F%B7#w%CC%83)*p,k\**(*m*). Substituting ( 26 ) and ( 28 ) into ( 33 ), we obtain

|  |  |
| --- | --- |
| , | ( 34 ) |

where \* indicates linear convolution with respect to the time index *n*, and

|  |  |
| --- | --- |
| . | ( 35 ) |

From ( 34 ), we know that *hp,k,pˊ,k*ˊ depends on (*p* - *pˊ*) rather than on *p* and *pˊ* separately. Substituting ( 34 ) into ( 32 ), we obtain

|  |  |
| --- | --- |
| . | ( 36 ) |

From ( 34 ), we also obtain

|  |  |
| --- | --- |
| . | ( 37 ) |

From ( 35 ), we get

|  |  |
| --- | --- |
| , | ( 38 ) |

where *wn,k* is the STFT representation of the synthesis window *w*(*n*) calculated with a decimation factor *Ls*=1. Equation ( 36 ) demonstrates that for a particular frequency-band index *k*, the temporal signal can be acquired by convolving the signal *xp,k*ˊ in each frequency-band *k*ˊ(*k*ˊ= 0,1,…,*N*-1) with its corresponding filter *hp,k,k*ˊ and subsequently adding up all the outputs. Here, the term for *k* = *k*ˊ is referred to as a band-to-band filter, and *k* ≠ *k*ˊ is referred to as a crossband filter, and crossband filters are employed to eliminate the aliasing effects resulting from subsampling. Note that ( 37 ) indicates that, in general, for fixed *k* and *k*ˊ, the filter *hp,k,k*ˊ has ⌈*N*/ *Ls* ⌉ *–* 1 non-causal coefficients. Hence, in echo cancellation applications, these coefficients must be taken into consideration. Extra delay of (⌈*N*/ *Ls* ⌉ *–* 1) *Ls* samples is typically introduced into the microphone signal to deal with this problem, as illustrated in [31] .

3.2. Band-to-band Filter as CTF Signal Model

In this paragraph, we will derive a CTF signal model for blind ATF estimation using band-to-band filters.

In a noise-free and reverberant environment, a speech signal transmits to microphones via the room effect. In the time domain, the received source image *y*(*n*) is specified by

|  |  |
| --- | --- |
| , | ( 39 ) |

where *s*(*n*) and *a*(*n*) represent the source signal and the RIR, respectively, with \* indicating linear convolution with respect to time index *n*. The RIR in ( 39 ) is often estimated using MTF in the STFT domain, as demonstrated by

|  |  |
| --- | --- |
| , | ( 40 ) |

where *yp,k* and s*p,k* represent the STFTs of their respective signals, while *ak* denotes the Fourier transform of the RIR *a*(*n*). In addition, *p* ∈ [1, *P*] refers to the frame index, *N* indicates the STFT window size, and *k* ∈ [0, *N–*1] represents the frequency index as in crossband filter. However, it is important to note that the approximation in ( 40 ) is accurate only if the length of the RIR *a*(*n*) is shorter than the STFT window size *N*. In real-world scenarios, numerous filter taps must be taken into account, numbering in the thousands, resulting in a destroyed approximation. Therefore, a significant increase in computational complexity and a slow convergence rate will occur. To overcome this problem, the crossband filter model is used in this study. From ( 36 ) the STFT coefficient *yp,k* is expressed as the sum of several convolutions between the STFT-domain source signal and the filter across the frequency bins, as follows:

|  |  |
| --- | --- |
| . | ( 41 ) |

Assuming *Ls* is the STFT frame step as stated above, if *Ls* is less than *N*, then *apˊ,k,k*ˊ becomes non-causal, with ⌈*N*/ *Ls* ⌉ *–* 1 non-causal coefficients. The number of causal filter coefficients is dependent on the reverberation time. For simpler notation, we assume that the filter index *p*ˊ ranges from 0 to *L –* 1, where *L* is the length of the filter. This requires shifting the non-causal coefficients to the causal component, which leads to a fixed delay shift of ⌈*N*/ *Ls* ⌉ *–* 1 of the frame index for the received microphone signal [31] . From ( 37 ) the STFT domain impulse response *apˊ,k,k*ˊ relates to the time domain impulse response *a*(*n*) by

|  |  |
| --- | --- |
| , | ( 42 ) |

which indicates the convolution with respect to the time index n evaluated at frame steps using ( 38 ). Note that for the remainder of this article, we will continue to refer to the analysis and synthesis windows in the STFT procedure as *w̃*(*n*) and *w*(*n*), respectively. To streamline the analysis, we employ the so called CTF approximation, which focuses exclusively on the band-to-band filters with *k* = *kˊ*, as described in the following:

|  |  |
| --- | --- |
| . | ( 43 ) |

Based on this, we are considering a version with multiple channels of *M* microphones

|  |  |
| --- | --- |
| , | ( 44 ) |

where *yi p,k* and *ai p,k* represent the *i*-th microphone signal and the corresponding CTF, respectively. Therefore, the source signals can be expressed in matrix form as follow:

|  |  |
| --- | --- |
| . | ( 45 ) |

Since the proposed algorithm functions on a frequency basis, the frequency index will be omitted henceforth for the sake of brevity. From ( 45 ) we can rewrite the matrix form with respect to frame index as follow:

|  |  |
| --- | --- |
| , | ( 46 ) |

where the bold symbols denote vectors or matrices and the subscript *d* denotes the delayed signal. Up to this point, we have derived the CTF signal model, which corresponds to equation ( 46 ).

# PROPOSED METHOD

This section presents a technique for estimating the ATF in a blind manner. It should be noted that the data available for analysis is limited to the positions of the microphones, the delayed microphone signal **y***d*,*p* (by ⌈N/ *Ls* ⌉ – 1 frames [31] ) and the pre-processed source signal **s***DAS,p*, which is obtained from the non-delayed microphone signal **y***p*. The position of the source is acquired via the TDOA estimated by GCC-PHAT. In particular, the pre-processed source signal **s***DAS,p* is derived through the application of the WPE algorithm and the DAS beamformer. It is noteworthy that three techniques for estimating the CTF coefficients are provided, namely the Wiener filter, RLS and Kalman adaptive filter. Furthermore, the parameters of all these filters were optimized using PSO and its enhanced version, as detailed in this thesis.

4.1. TDOA-based Source Localization

TDOA is the difference in the arrival times of the emitted signal received at a pair of microphones. Upon receipt of the signal by the microphones, an estimation of TDOA can be made by means of GCC-PHAT. Subsequently, the distance difference between the source and the two microphones can be obtained by multiplying the known propagation speed. Finally, the distance difference can be used to locate the source position via the CLS algorithm.

4.1.1. GCC-PHAT for TDOA estimation

A signal originating from a distant source at two spatially distinct sensors can be mathematically represented as

|  |  |
| --- | --- |
| , | ( ) |

where the signal received at the first microphone is designated as , while the TDOA between the first and second microphones is designated as . It is necessary to transform the two time-domain signals into the frequency domain individually. Subsequently, the cross spectrum can be obtained as follows:

|  |  |
| --- | --- |
| , | ( ) |

where the Fourier transforms of  and , respectively, are represented as and . The phase transformation weighting scheme, as illustrated in ( 49 ), can be employed to achieve unity gain for each frequency component while preserving the phase data, which contains the actual delay information.

|  |  |
| --- | --- |
|  | ( ) |

The aforementioned result is then transformed to the time domain with the objective of obtaining a correlation function as follows:

|  |  |
| --- | --- |
| . | ( 50 ) |

In theory, upon returning  to the time domain, we should obtain a unit impulse function. This result is based on the following fact:

|  |  |
| --- | --- |
| . | ( 51 ) |

It can thus be concluded that the peak of the correlation function will indicate the delay time. Nevertheless, in order to enhance the precision of the results, it is possible to employ an interpolation method based on the convolution of a Sinc function and a correlation function.

|  |  |
| --- | --- |
|  | ( 52 ) |

The delay time can be calculated as follows:

|  |  |
| --- | --- |
| . | ( 53 ) |

4.1.2. TDOA Measurement Model

|  |
| --- |
|  |
| Figure 1 The sound sources propagation model |

The sound propagation model is depicted in Figure 1. It is assumed that *S* represents the source, that  are the microphones, and that the reference microphone is designated as . In accordance with the stipulations of section 4.1.1, the TDOA can be estimated by utilizing the GCC-PHAT algorithm. Consequently, the distance between the source and the microphone, and the distance between the source and the reference microphone, can be expressed as

|  |  |
| --- | --- |
| , | ( 54 ) |

where  represents the distance between the source and the *m-*th microphone, while  denotes the distance between the source and the reference microphone. The symbol  represents the TDOA between the source and the *m-*th microphone, while *c* represents the speed of sound. Finally,  represents Gaussian white measurement noise with a variance of . We posit that the source coordinate is , the *m*-th microphone coordinate is , and the reference microphone is placed at the origin. Consequently, by disregarding the measurement noise, ( 54 ) can be rewritten as

|  |  |
| --- | --- |
| . | ( 55 ) |

Subsequently, the second item in ( 55 ) is defined as *R* as follows:

|  |  |
| --- | --- |
| . | ( 56 ) |

Finally, we can convert the system of linear equations in ( 55 ) to matrix form as follows:

|  |  |
| --- | --- |
| , | ( 57 ) |

where

|  |  |
| --- | --- |
| . | ( 58 ) |

Subsequently,  may be estimated utilizing the standard least-squares (LS) method, as follows:

|  |  |
| --- | --- |
| , | ( 59 ) |

where  represents a vector of optimization variables. In order to achieve enhanced performance, it is necessary to incorporate considerations of measurement error. Assuming a high signal-to-noise ratio (SNR) of the measurement, the squared distance can be expressed as

|  |  |
| --- | --- |
| . | ( 60 ) |

Consequently, the discrepancy between the actual and the measured squared distances is

|  |  |
| --- | --- |
| . | ( 61 ) |

In vector form, it can be expressed as

|  |  |
| --- | --- |
| . | ( 62 ) |

The covariance matrix of the disturbance is therefore of the form

|  |  |
| --- | --- |
| , | ( 63 ) |

where

|  |  |
| --- | --- |
| . | ( 64 ) |

Finally, the weighting matrix  can be employed to formulate the weighted least-square localization problem as follows:

|  |  |
| --- | --- |
| . | ( 65 ) |

4.1.3. Constrained Least Squares (CLS) method

It is important to note that, in ( 57 ), the variable *R* is dependent on the variables *x*, *y*, and *z*. Therefore, it is necessary to apply a constraint in order to satisfy the basic relationship.

|  |  |
| --- | --- |
| . | ( 66 ) |

As a result, we adopt the method of Lagrange multipliers as a strategy for identifying the local minimum of a function subject to equation constraints. This is achieved by:

|  |  |
| --- | --- |
| . | ( 67 ) |

The solution of **錯誤! 找不到參照來源。** can be readily obtained by applying the partial derivative with respect to .

|  |  |
| --- | --- |
| . | ( 68 ) |

From ( 68 ), it can be seen that the sole variable is , which allows us to modify CLS to a root-finding problem as follows:

|  |  |
| --- | --- |
| . | ( 69 ) |

Nevertheless, the real roots of ( 69 ) are likely not singular. In the event that multiple real roots are identified, a number of solutions may be obtained by substituting each root into ( 68 ). Subsequently, the optimal solution is selected by minimizing the objective function in ( 65 ). To date, we have successfully identified the genuine location of the source.

4.2. Pre-processing

In order for the subsequent CTF estimation algorithm to function effectively, it is necessary to have access to a source signal that is free from contamination and echoes. However, in practical applications, obtaining a clean source signal is often challenging. Consequently, this thesis employs the Weighted Prediction Error (WPE) method for dereverberation, as outlined in [13] . Subsequently, the Delay and Sum (DAS) beamformer [14] is employed with the source location obtained from section 4.1. in order to extract a clean source signal from the WPE outputs of all channels.

4.2.1. WPE

In the event that a single speech source is captured by *M* microphones, it is possible to rewrite ( 39 ) as follows:

|  |  |
| --- | --- |
| , | ( 70 ) |

where *m* and *La* represent the ordinal numbers of the *m*-*th* microphone and the length of the RIR. The reverberant signal *ym*(*n*) in ( 70 ) is comprised of three distinct components: a direct signal, early reverberation, and late reverberation. It is a common practice to take the first two components as the desired signal, which is denoted by *dm*(*n*). Concurrently, the late reverberation is designated as the signal to be eliminated and is denoted by *rm*(*n*). The relationship between these signals can be expressed as follows:

|  |  |
| --- | --- |
| , | ( 71 ) |

where

|  |  |
| --- | --- |
| , | ( 72 ) |

where *D* is the sample index that distinguishes the RIR into the early and late reverberation parts. This index is subsequently referred to as the prediction delay. From now on, we assume that there two microphones, namely *M* = 2, for the sake of simplicity. If the RIRs *am*(*n*) in different channels do not share common zeros, the relationship between speech signal and the microphone signals in ( 70 ) can be rewritten (stepwise derivation is shown in [32] ) as

|  |  |
| --- | --- |
| , | ( 73 ) |

where

|  |  |
| --- | --- |
| , | ( 74 ) |

where **c***m* and *Lc* indicate the vector of regression coefficients and the regression order, respectively, and *LH* equals *Lc*+ *La* – 1.

Using the estimated vector of regression coefficients and following ( 73 ), it is possible to acquire the desired signal as follow:

|  |  |
| --- | --- |
| . | ( 75 ) |

Hence, the dereverberation can be achieved by obtaining a suitable estimated vector of regression coefficients from the microphone signals. Because is completely determined independently of , in the following, we disregard the optimization of without loss of generality. The resultant optimization algorithm can be summarized (stepwise derivation is shown in [13] ) as follow.

|  |
| --- |
| Algorithm 1 WPE |
| Input: *y*1(*n*), **y**(*n*)   1. Initialize as  |  |  | | --- | --- | | , | ( 76 ) |   where *Lf* is length of short time frame and *μ* > 0 is a certain lower bound to avoid zero division.   1. Repeat the following steps until convergence. 2. Update as follows:  |  |  | | --- | --- | | , | ( 77 ) |   where + denotes the pseudo inverse and   |  |  | | --- | --- | |  | ( 78 ) |      |  |  | | --- | --- | | , | ( 79 ) |   where *τ* is the largest sample index of the microphone signal.   1. Update as . 2. Update as follow:  |  |  | | --- | --- | | . | ( 80 ) | |

It is worth noting that we will refer to the WPE output of the *m*-*th* channel as from this point forward.

4.2.2. DAS Beamformer

Once the de-reverberated microphone signal from WPE is obtained, clean source signal extraction through DAS beamformer can be executed. First, we convert to the STFT domain, resulting in . Here, *p* still represents the frame index as previously explained. Secondly, we calculate the beamforming weight of the DAS beamformer as follow:

|  |  |
| --- | --- |
| , | ( 81 ) |

where *к* denotes the wave number and the distance between the source and the *m*-th microphone, designated by *Rm*, are computed through the location of the source obtained in section 4.1.. Finally, the inner product is performed between the weight and to obtain the clean source signal as follows:

|  |  |
| --- | --- |
| , | ( 82 ) |

where *H* denotes Hermitian transpose.

4.3. Wiener Filtering Approach

For the first technique, the Wiener-based derivations are employed to estimate the matrix of CTF coefficients **A**. This approach minimizes the mean square error as follow:

|  |  |
| --- | --- |
| , | ( 83 ) |

where E[·] denotes the expectation with respect to the frames. Therefore, ( 83 ) can be rewritten as

|  |  |
| --- | --- |
| , | ( 84 ) |

where *tr*{·} denotes the matrix trace, and the associated covariance matrices is

|  |  |
| --- | --- |
| . | ( 85 ) |

By taking the derivative of ( 84 ) with respect to , we obtain

|  |  |
| --- | --- |
| . | ( 86 ) |

The optimal Wiener solution can be obtained as

|  |  |
| --- | --- |
| . | ( 87 ) |

In practical implementation, instead of the expectation, the recursive averaging is adopted to obtain **Rsy** and **Rss** as given by

|  |  |
| --- | --- |
| , | ( 88 ) |

where α denotes the forgetting factor for the recursive averaging process. The Wiener filtering approach can be summarized as follows.

|  |
| --- |
| Algorithm 2CTF estimation using Wiener filtering |
| Input: **y***d,p*, **s***DAS,p*   1. Initialize forgetting factor *α* and covariance matrices as 2. For each instant of frame, *p* = 1, 2, …, compute  |  |  | | --- | --- | | . | ( 89 ) | |

It should be noted that the estimated matrix of CTF coefficients changes depending on the processed frame, and the accuracy improves as the number of processed frames increases.

4.4. RLS Approach

For the second technique, the CTF coefficients matrix is estimated through the application of the adaptive filter algorithm. It is worth noting that the RLS algorithm optimization process discussed in [16] is currently being conducted in the complex domain. The RLS algorithm aims to minimize the sum of the weighted error norm square as

|  |  |
| --- | --- |
| , | ( 90 ) |

where *p* represents both the adaptation iteration and the frame index, while λ represents the forgetting factor multiplied by the square of the error norm concerning the iteration. Guided by the objective function articulated in ( 90 ), the RLS algorithm is employed for the estimation of the CTF coefficients matrix . The RLS approach can be succinctly summarized in the subsequent algorithmic routine.

|  |
| --- |
| Algorithm 3 CTF estimation using RLS |
| Input: **y***d,p*, **s***DAS,p*   1. Initialize RLS forgetting factor *λ*, weight and inverse of correlation matrix as   , where *ε* is a small positive constant   1. For each instant of frame, *p* = 1, 2, …, compute  |  |  | | --- | --- | | . | ( 91 ) | |

It should be noted that the estimated matrix of CTF coefficients changes depending on the processed frame, and the accuracy also improves as the number of processed frames increases.

4.5. Kalman Algorithm Adaptive Filtering Approach

In the third technique, the CTF coefficient matrix is estimated by applying the Kalman filter. It is worth noting that in this thesis we adapt the Kalman filter as an adaptive filter instead of using it as a state space control filter. Despite this modification, the primary concept remains the same. The process equation of the stationary Kalman adaptive filter of each microphone without process noise is described as

|  |  |
| --- | --- |
| , | ( 92 ) |

where ∈ ℂLх1 signifies the optimal weight vector and has a connection with the CTF coefficients matrix as

|  |  |
| --- | --- |
| , | ( 93 ) |

where denotes the *m*-*th* row of **A***p*. The measurement equation of stationary Kalman adaptive filter of each microphone is described as

|  |  |
| --- | --- |
| , | ( 94 ) |

where denotes the measurement noise for each microphone, and

|  |  |
| --- | --- |
| , | ( 95 ) |

where is the covariance of measurement noise.

Using the process and measurement equations outlined in ( 92 ) and ( 94 ), the Kalman gain can be derived by minimizing the error covariance matrix [17] . The subsequent algorithmic routine provides a succinct summary of the stationary Kalman adaptive filter approach.

|  |
| --- |
| Algorithm 4 CTF estimation using stationary Kalman adaptive filtering |
| Input: , **s***DAS,p*   1. Initialize estimated Kalman weight, error covariance matrix, Kalman gain and measurement noise covariance as   , where *η* and *ρ* is a small positive constant   1. For each microphone, *m* = 1, 2, …,   For each instant of frame, *p* = 1, 2, …, compute   |  |  | | --- | --- | | . | ( 96 ) | |

It is easy to observe that the estimated matrix of CTF coefficients shows a variability depending on the processed frame, and the accuracy also improves as the number of processed frames increases.

In this section we have only introduced the stationary Kalman adaptive filter. However, the non-stationary version is useful when the location of the source changes with time. The derivation of the non-stationary version is simply obtained by introducing process noise into the process equation, so we omit it in this thesis for the sake of simplicity. Nevertheless, it will be used in our moving source simulation cases.

4.6. ATF Reconstruction

Once the matrix of CTF coefficients has been estimated from the three approaches mentioned earlier, we can proceed with producing the ATFs. Our first step is to generate a unit pulse sequence, which experiences a delay of (*L* – 1) *Ls* points. Subsequently, it is transformed into the STFT domain, resulting in *δp,k* as follows:

|  |  |
| --- | --- |
| . | ( 97 ) |

It is obvious that the magnitude in different frame index *p* is a constant along the frequency axis, depending on the analysis window used. Finally, the estimated CTF coefficients are convolved with it to give the following signal:

|  |  |
| --- | --- |
| , | ( 98 ) |

where *p* ∈ [0, *PATF*]. The estimated RIRs , *n* ∈ [0, *NATF*], can be obtained by applying the inverse STFT to . Subsequently, the estimated ATFs, represented by vector with each element corresponding to different frequency bins, can be obtained by performing a fast Fourier transform (FFT) on the estimated RIRs . is expressed as

|  |  |
| --- | --- |
| . | ( 99 ) |

4.7. Parameters Optimization

While the algorithms outlined above showcase promising outcomes in simulations, it is imperative to recognize the considerable effect that parameters within these algorithms can have on the outcomes. To optimize the performance of the algorithms proposed, we utilize Particle Swarm Optimization (PSO) and its advanced versions [19] to optimize the parameters that are involved.

4.7.1. PSO

The PSO algorithm is a swarm intelligent optimization technique inspired by the flocking of birds and schooling of fish [18] . PSO represents each particle's position as a candidate solution during the exploration of a *U*-dimensional space. At the *t*-*th* update iteration, one particle *j* among the *J* particles in the population is characterized by its position and velocity as follows:

|  |  |
| --- | --- |
| . | ( 100 ) |

Let the fitness function  be the one that is required to be minimized. The function accepts a candidate solution in the form of a real vector and produces a real number that represents the fitness value of the given candidate solution. In our case, the candidate solution corresponds to the parameters in our proposed algorithms, and the fitness function can be described as follows:

|  |  |
| --- | --- |
| , | ( 101 ) |

where represents the estimated CTF coefficients matrix obtained from any one of the three methods when the parameters are specified. After calculating the fitness value of the entire population, *pbestj*(*t*) and *gbest*(*t*) are updated, which are the personal best position of the *j*-*th* particle and the global best position in the population, respectively.

|  |  |
| --- | --- |
|  | ( 102 ) |

The velocity and position are then updated using the formulas as below:

|  |  |
| --- | --- |
| , | ( 103 ) |

where *win* represents the inertia weight, *r*1 and *r*2 are random variables that fall within the interval [0, 1] and c1 and c2 denote two positive acceleration coefficients. It is noteworthy that the update process will persist as long as the maximum iteration limit *Tmax* has not been reached. The PSO algorithm's entire process is presented in Figure 2.

|  |
| --- |
|  |
| Figure 2 Block diagram of PSO |

4.7.2. ASPSO

Although PSO is widely used in the optimization process, it remains limited in its ability to address complicated optimization problems, including premature convergence and insufficient balance between global exploration and local exploitation. To mitigate these challenges, a novel hybrid PSO algorithm using an adaptive strategy (ASPSO) has been developed [19] . It includes four main modifications, namely: inertia weight with chaotic, elite and dimensional learning strategies, adaptive position update strategy and competitive substitution mechanism. These modifications are explained in the following sections.

The inertia weight *win* plays a key role in harmonizing exploration and exploitation within the search progress. Therefore, the choice of the inertia weight is important. While a linear inertia weight is commonly used, the majority of real-world practical scenarios involve complex non-linear systems. Taking advantage of the randomness, ergodicity and sensitivity inherent in chaotic maps, the C-PSO algorithm incorporates a non-linear approach to adjusting the inertia weight [33] . The formula for calculating inertia weight is

|  |  |
| --- | --- |
| , | ( 104 ) |

where *Cin*, *wmax*, *wmin* and *Tmax* denote a small positive integer, maximum inertia weight, minimum inertia weight and maximum iteration, respectively.

The basic PSO uses personal and global learning strategies to control the velocity and position updates of the particles. Specifically, all particles use their collective best experiences (*pbestj*(*t*) and *gbest*(*t*)) to accelerate solution progress. However, this approach can lead to trapping in local optimal when dealing with multimodal features. To mitigate this challenge, [19] introduce elite and dimensional learning strategies. In the elite learning strategy, particles learn from exceptional individuals to increase the diversity of the population. Throughout the search, each particle learns from four personal best positions of different particles *pbestj*(*t*) randomly selected from the population. Subsequently, the personal best particle *j* is compared with the above four particles, and the particle with the best fitness value is retained as the new personal best (*Fpbestj*(*t*)). The learning strategy is expressed as

|  |  |
| --- | --- |
| . | ( 105 ) |

An excessive focus on *gbest*(*t*) can lead to a rapid diversity in population. To mitigate this potential problem, [19] use the dimensional learning method. By facilitating communication between particles in the dimensional aspect, the mean value provides complementary information, thereby increasing diversity and improving search efficiency. A global particle, denoted *Mpbest*(*t*), is defined as

|  |  |
| --- | --- |
| . | ( 106 ) |

Finally, the velocity update equation is changed to:

|  |  |
| --- | --- |
| . | ( 107 ) |

Conventional PSO faces the challenge of achieving an effective balance between global exploration and local exploitation during the search process. The position update law induces particles to consistently converge to their previously determined optimal positions, thereby limiting their ability to explore neighborhoods around the known optimal solution. In response to this constraint, a spiral mechanism has been introduced as a local search operator in the vicinity of the known optimal solution region [34] . Building on this inspiration, an adaptive position update strategy that generates particle positions by dynamically orchestrating a balance between local exploitation and global exploration is proposed in [19] . This strategy is articulated by

|  |  |
| --- | --- |
| , | ( 108 ) |

where *Dj* represents the distance between the current best position and the *j*-*th* particle. The parameter *b* serves as a constant that determines the shape of the logarithmic spiral, and *l* is a random number in the range [-1,1]. During each iteration, a ratio *βj*(*t*) is calculated by evaluating the fitness value of the current particle in relation to the average fitness value. If *βj*(*t*) is small, indicating that the particle is close to the optimal position, there is a need to increase its local exploitation capability. Conversely, if the particle is in a suboptimal position, an update is implemented to increase its global exploration capability, thereby mitigating premature convergence.

Finally, a competitive substitution mechanism is introduced to enhance the performance of PSO [19] . In each iteration, the worst-performing particle is identified and replaced, as defined by

|  |  |
| --- | --- |
| , | ( 109 ) |

where is a random number. During the search process, all particles in the population acquire knowledge from the global best particle *gbest*(*t*). Therefore, *gbest*(*t*) significant influences the entire population. In a complex search environment, if *gbest*(*t*) becomes trapped in a local optimum, the remaining particles tend to converge towards the suboptimal region, leading to premature convergence. Accordingly, a perturbation strategy is built into ASPSO to facilitate the escape of *gbest*(*t*) from local optimal. To minimize the time spent on unfavorable directions, a condition is set to trigger the perturbation strategy if *gbest*(*t*) fails to update its value after five iterations. The perturbation strategy is described as follows:

|  |  |
| --- | --- |
| , | ( 110 ) |

where is a random number.

The ASPSO algorithm's entire process is presented in Figure 3.

|  |
| --- |
|  |
| Figure 3 Block diagram of ASPSO |

4.8. Summary of Proposed Method

Table 1 summarizes the method proposed in this thesis, describing the resulting output signals obtained at each stage.

Table 1 Flow chart of our proposed method

|  |
| --- |
| **Flow chart of our proposed method** |
| Step 1: Acquire the microphone signal and delay it to get  Step 2: Do TDOA-based source localization to obtain source location  Step 3: Do WPE followed by DAS to obtaining **s***DAS,p*  Step 4: Estimate the CTF coefficients matrix via one of the algorithms below   1. Algorithm 2 CTF estimation using Wiener filtering 2. Algorithm 3 CTF estimation using RLS 3. Algorithm 4 CTF estimation using stationary Kalman adaptive filtering   Step 5: Do ( 97 ) ~ ( 99 ) to obtain estimated RIRs or ATFs  Step 6: Filter parameters optimization through PSO or ASPSO.  Step 7: Applications: MINT for dereverberation, MPDR beamformer for speech enhancement and TIKR for source separation. |

# SIMULATIONS

A CTF-based blind ATF estimation problem has been proposed and motivated, with the aim of achieving fast convergence, adaptivity and low computational complexity. The proposed solution is comprised of three distinct approaches, which have been developed in the preceding sections. For purposes of comparison, our approach has also been contrasted with the state-of-the-art BSI method, namely MCLMS. The simulation cases include both fixed and moving sources. This chapter also includes optimizations of filter parameters and applications using estimated ATF or RIR.

5.1. Fixed Source Location Cases

Three distinct room settings have been devised for the generation of different reverberation times (RIRs) utilizing the RIR Generator [35] , with the specifications of each room listed in Table 2. It is important to note that the system employs a hybrid compact-distributed array comprising eight microphones in cuboid distributed part, with the number of microphones spaced 0.02m utilized in the compact part, namely ULA, contingent upon the reverberation time. At 16 kHz, speech signals were sampled and employed as sources for generating microphone signals, which were convolved with the reverberation impulse responses (RIRs) of the ground truth. For each room's layout, we present it in Figure 4.

Table Specifications of room settings for different RIRs

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Settings  specifications | Range of *T*60 (sec) | Dimensions of the room (m) | Number of microphones of ULA | First sensor location of ULA (m) | Dimensions and first sensor location of distributed array (m) | Source location (m) |
| Room1 | 0.01 | 0.3 × 0.4 ×0.3 | 30 | 0.1, 0.1, 0.2 | 0.1×0.1×0.1  0.15, 0.25, 0.15 | 0.2, 0.3, 0.2 |
| Room2 | 0.1 | 3 × 3×2.5 | 10 | 1.1, 1, 1 | 1 × 1 ×1  1, 1, 1 | 1.7, 1.8, 1.3 |
| Room3 | 0.2~1.6 | 5× 6×2.5 | 10 | 2.1, 2, 1 | 1 × 1 ×1  1, 1, 1 | 2.1, 2.15, 1.1 |

|  |
| --- |
|  |
| Figure Configurations of the room for fixed source location |

In section 5.1., the values of the free parameters α, λ, ε, η and ρ were consistently fixed at 0.999, 0.99, 0.01, 0.5 and 0.001, respectively, as they were found to be appropriate for all conditions. The magnitude and phase of the estimated ATF for all frequency bins and the magnitude of the estimated RIR with several reverberation times chosen are compared to their ground truth values and displayed in Figure 5. Nevertheless, when the reverberation time exceeds 0.1 seconds, it becomes challenging for MCLMS to converge due to the prolonged RIR. Consequently, MCLMS simulations are only conducted with reverberation times below 0.1. Besides, it is important to note that in the case of blind estimation, there is an inevitable equalization problem, whereby there will be a scale gap between the estimated RIR and the ground-truth RIR. To address this issue, the estimated RIR is rescaled using the ratio calculated as the maximum absolute magnitude of the ground-truth RIR divided by the maximum absolute magnitude of the estimated RIR. Figure 5 reveals a minimal absolute magnitude and phase errors across all frequency bins and a remarkable correspondence between the magnitude of the estimated RIR and its ground-truth counterparts, which fulfils the desired outcome. Table 3 and Figure 6 illustrate the normalized root mean square projection misalignment (NRMSPM) of the estimated ATFs for all algorithms. It is important to note again that MCLMS simulations are only conducted with reverberation times below 0.1. Consequently, the NRMSPM values are set to zero for those reverberation times. Furthermore, the algorithms with the lowest NRMSPM at each reverberation time are indicated in red. The NRMSPM is defined as follows:

|  |  |
| --- | --- |
| , | ( ) |

where *N* represents the number of Monte Carlo runs, **g** denotes a long vector that is connected using the ground-truth RIR of each channel, (•)(*i*) denotes a value obtained from the *i*-th run, and the projection misalignment vector  is depicted as follows :

|  |  |
| --- | --- |
| , | ( ) |

where denotes a long vector that is connected using the estimated RIR of each channel.

|  |  |
| --- | --- |
|  |  |
| a-1 Wiener filter | |
|  |  |
| a-2 RLS | |
|  |  |
| a-3 Kalman filter | |
|  |  |
| a-4 MCLMS | |
| 1. *T*60 = 0.01 | |
|  |  |
| b-1 Wiener filter | |
|  |  |
| b-2 RLS | |
|  |  |
| b-3 Kalman filter | |
| 1. *T*60 = 0.5 | |
|  |  |
| c-1 Wiener filter | |
|  |  |
| c-2 RLS | |
|  |  |
| c-3 Kalman filter | |
| 1. *T*60 = 1.6 | |
| Figure Magnitude and phase of the estimated ATF and magnitude of the estimated RIR of all algorithms with several chosen *T*60 | |

Table 3 NRMSPM of the estimated ATFs for all algorithms in various reverberation times

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| method  *T*60 (s) | Wiener | RLS | Kalman | MCLMS  (baseline) |
| 0.01 | -9.3472 | -9.0147 | -9.3561 | -7.4119 |
| 0.1 | -7.1719 | -6.1848 | -7.1623 | -2.98E-07 |
| 0.2 | -14.2432 | -14.9637 | -13.5157 | 0 |
| 0.3 | -14.3913 | -14.8531 | -13.8433 | 0 |
| 0.4 | -14.7163 | -14.793 | -14.3051 | 0 |
| 0.5 | -14.3667 | -14.5819 | -13.8722 | 0 |
| 0.6 | -14.3403 | -14.7172 | -13.8901 | 0 |
| 0.7 | -14.2824 | -14.4914 | -13.9392 | 0 |
| 0.8 | -14.1173 | -14.4959 | -13.7197 | 0 |
| 0.9 | -13.7846 | -13.9609 | -13.4715 | 0 |
| 1.0 | -13.8204 | -13.9497 | -13.4665 | 0 |
| 1.1 | -13.6486 | -13.4545 | -13.3825 | 0 |
| 1.2 | -13.4506 | -13.6967 | -13.0737 | 0 |
| 1.3 | -13.0662 | -13.0599 | -12.7338 | 0 |
| 1.4 | -12.6869 | -12.7366 | -12.3377 | 0 |
| 1.5 | -12.952 | -12.5737 | -12.8581 | 0 |
| 1.6 | -12.767 | -12.2751 | -12.7297 | 0 |

|  |
| --- |
|  |
| Figure NRMSPM of the estimated ATFs for all algorithms in various reverberation times |

5.2. Fixed Source Location with Parameters Optimization

5.3. Applications

5.4. Moving Source Location Cases

# EXPERIMENTS

6.1. Experiment Settings and Parameters

6.2. Experiment Results and Discussions

# CONCLUSIONS AND FUTURE WORK

* 1. Conclusions
  2. Future Work

# REFERENCES

1. J. Benesty, T. Gänsler, D. R. Morgan, M. M. Sondhi, and S. L. Gay, “Advances in Network and Acoustic Echo Cancellation,” *New York: Springer.,* 2001.
2. Y. Huang, J. Benesty, and J. Chen, “A blind channel identification-based two-stage approach to separation and dereverberation of speech signals in a reverberant environment,” *IEEE Trans. Speech Audio Process.,* vol. 13, no. 5, pp. 882–895, 2005.
3. S. Araki, R. Mukai, S. Makino, T. Nishikawa, and H. Saruwatari, “The fundamental limitation of frequency domain blind source separation for convolutive mixtures of speech,” *IEEE Trans. Audio, Speech, Lang. Process.,* vol. 14, no. 3, pp. 774–784, May 2006.
4. S. Gannot, D. Burshtein, and E. Weinstein, “Signal enhancement using beamforming and nonstationarity with applications to speech,” *IEEE Trans. Audio, Speech, Lang. Process.,* vol. 49, no. 8, pp. 1614–1626, Aug. 2001.
5. Y. Huang and J. Benesty, “Adaptive multi-channel least mean square and Newton algorithms for blind channel identification,” *Signal Processing*, vol. 82, no. 8, pp. 1127–1138, Aug. 2002.
6. Y. Huang and J. Benesty, “A class of frequency-domain adaptive approaches to blind multichannel identification,” *IEEE Trans. Signal Processing*, vol. 51, no. 1, pp. 11–24, Jan. 2003.
7. M. K. Hasan, J. Benesty, P. A. Naylor, and D. B. Ward, “Improving robustness of blind adaptive multichannel identification algorithms using constraints,” *Proc. European Signal Processing Conference (EUSIPCO),* 2005.
8. Y. Avargel and I. Cohen, “On multiplicative transfer function approximation in the short-time Fourier transform domain,” *IEEE Signal Processing Letters.,* vol. 14, no. 5, pp. 337–340, 2007.
9. Y. Avargel and I. Cohen, “System identification in the short-time Fourier transform domain with crossband filtering,” *IEEE Trans. Audio, Speech, Lang. Process.,* vol. 15, no. 4, pp. 1305–1319, 2007.
10. R. Talmon, I. Cohen and S. Gannot, “Relative transfer function identification using convolutive transfer function approximation,” *IEEE Trans. Audio, Speech, Lang. Process.,* vol. 17, no. 4, pp. 546–555, 2009.
11. B. Van Den Broeck, A. Bertrand, P. Karsmakers, B. Vanrumste, H. Van hamme and M. Moonen, “Time-domain generalized cross correlation phase transform sound source localization for small microphone arrays,” *2012 5th European DSP Education and Research Conference (EDERC).,*pp.76-80, 2012.
12. T. Li, Z. Deng, G. Wang and J. Yan, "Time Difference of Arrival Location Method Based on Improved Snake Optimization Algorithm, “*2022 IEEE 8th International Conference on Computer and Communications (ICCC) .,* pp. 482-486, 2022.
13. T. Nakatani, T. Yoshioka, K. Kinoshita, M. Miyoshi and B. -H. Juang, “Speech Dereverberation Based on Variance-Normalized Delayed Linear Prediction,” *IEEE Trans. Audio, Speech, Lang. Process.,* vol. 18, no. 7, pp. 1717–1731, 2010.
14. Harry L. Van Trees, “Optimum array processing: Part Iv of detection, estimation, and modulation theory,” *New York: Wiley*, 2002*.*
15. J. Benesty, S. Makino, J. Chen, Y. Huang, and S. Doclo, “Study of the Wiener filter for noise reduction,” *Speech enhancement,* pp. 9-41, 2005.
16. S. A. U. Islam and D. S. Bernstein, “Recursive Least Squares for Real-Time Implementation,”IEEE Control Systems Magazine*,* vol. 39, no. 3, pp. 82-85, June 2019.
17. R. E. Kalman, “A New Approach to Linear Filtering and Prediction Problems,” *Journal of Basic Engineering,* 1960.
18. J. Kennedy, and R. Eberhart, “Particle swarm optimization,” *Proceedings of ICNN’95-International Conference on Neural Networks, IEEE,* pp. 1942-1948, 1995.
19. Rui ang, Kuangrong Hao, Lei Chen, Tong Wang, and Chunli Jiang, “A novel hybrid particle swarm optimization using adaptive strategy,” Information Sciences, vol. 579, pp. 231-250, 2021.
20. Y. Huang and J. Benesty, “Adaptive blind channel identification: Multi-channel least mean square and Newton algorithms,” *2002 IEEE International Conference on Acoustics, Speech, and Signal Processing,* pp. II-1637-II-1640, 2002.
21. M. Miyoshi, and Y. Kaneda, “Inverse filtering of room acoustics,” *IEEE Trans. Audio, Speech, Lang. Process.,* vol. 36, no. 2, pp. 145-152, Feb. 1988.
22. R. Marxer and J. Janer, “A Tikhonov regularization method for spectrum decomposition in low latency audio source separation,” *2012 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP).,* pp. 277-280, 2012.
23. Q. Trong The and N. Thi Thu Dung, “A Speech Enhancement in Diffuse Noise Field using Minimum Variance Distortionless Response Filter,”  *2022 Conference of Russian Young Researchers in Electrical and Electronic Engineering (ElConRus).,* pp. 1395-1398, 2022.
24. A. W. Rix, J. G. Beerends, M. P. Hollier, and A. P. Hekstra, “Perceptual evaluation of speech quality (PESQ)-a new method for speech quality assessment of telephone networks and codecs,”*IEEE International Conference on Acoustics, Speech, and Signal Processing. Proceedings,* vol. 2,pp. 749-752, 2001.
25. E. Vincent, R. Gribonval, and C. Fevotte, “Performance measurement in blind audio source separation,” *IEEE Trans. Audio, Speech, Lang. Process.,* vol. 14, no. 4, pp. 1462–1469, 2006
26. S. Haykin, “Adaptive Filter Theory,” *Englewood Cliffs, NJ: Prentice-Hall,*, 1996.
27. M. R. Portnoff, “Time-frequency representation of digital signals and systems based on short-time Fourier analysis,” *IEEE Trans. Signal Process.,* vol. ASSP-28, no. 1, pp. 55–69, Feb. 1980.
28. S. Farkash and S. Raz, “Linear systems in Gabor time-frequency space,” *IEEE Trans. Signal Process.,* vol. 42, no. 3, pp. 611–617, Jan. 1998.
29. J. Wexler and S. Raz, “Discrete gabor expansions,” *Signal Process.,* vol. 21, pp. 207–220, Nov. 1990.
30. S. Qian and D. Chen, “Discrete Gabor transform,” *IEEE Trans. Signal Process.,* vol. 41, no. 7, pp. 2429–2438, Jul. 1993.
31. W. Kellermann, “Analysis and design of multirate systems for cancellation of acoustical echoes, “ *International Conference on Acoustics, Speech, and Signal Processing,* vol. 5, pp. 2570-2573, 1988.
32. D. Gesbert, and P. Duhamel, “Robust blind channel identification and equalization based on multi-step predictors,” *International Conference on Acoustics, Speech, and Signal Processing,* pp. 3621–3624, 1997.
33. H. A. Hefny, and S. S. Azab, “Chaotic particle swarm optimization,” *The 7th International Conference on Informatics and Systems (INFOS),* pp. 1-8,2010.
34. Seyedali Mirjalili, and Andrew Lewis, “The whale optimization algorithm,” *Advances in Engineering Software,* vol. 95, pp.51-67, 2016.
35. Emanuël Habets, “Room Impulse Response Generator,” *Internal Report,* pp. 1-17, 2006*.*