國 立 清 華 大 學

碩 士 論 文 計 劃 書

使用卷積傳遞函數進行聲學傳遞函數盲估計以應用於去混響

Blind estimation of acoustic transfer functions with application to dereverberation using convolutive transfer functions

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 二 年 十一 月

摘要

雖然聲學傳遞函數（Acoustic Transfer Functions）在各種陣列信號處理應用中通常優於相對傳遞函數（Relative Transfer Functions），但源信號的不可取用性對於獲得可靠的聲學傳遞函數估算構成了重大挑戰。為應對這一問題，我們提出了一種新穎的基於卷積傳遞函數（Convolutive Transfer Functions）的聲學傳遞函數盲估算方法。首先，使用加權預測誤差算法（Weighted Prediction Error）和延遲和加總（Delay and Sum）波束成形器來獲得聲源位置的目標源信號之初始估算。隨後，使用維納濾波器或卡爾曼濾波器計算卷積傳遞函數之係數，並通過粒子群優化（Particle Swarm Optimization）來最佳化濾波器中的參數。為了得到聲學傳遞函數的脈衝響應，將單位脈衝序列的短時距傅立葉變換（Short-time Fourier Transform）與卷積傳遞函數係數進行卷積，然後應用逆短時距傅立葉變換（Inverse Short-time Fourier Transform）來獲得它。為了證明我們提出的聲學傳遞函數估算技術的有效性，我們以多輸入/輸出逆定理（Multiple Input/Output Inverse Theorem）的混響消除應用作為示例。此應用需要準確的聲學傳遞函數估算。使用 30 個元素的線性陣列進行的模擬結果顯示，我們的方法產生的聲學傳遞函數估算與真實的房間脈衝響應非常相符。此外，經多輸入/輸出逆定理去混響的信號在兩個客觀性能指標中都與乾淨的源信號表現出顯著的相似性。

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，多輸入/輸出逆定理

ABSTRACT

While Acoustic Transfer Functions (ATFs) often outperform Relative Transfer Functions (RTFs) in various array signal processing applications, the unavailability of the source signal poses a significant challenge for obtaining reliable ATF estimates. In response to this issue, we introduce an innovative approach for blind ATF estimation, which is based on Convolutive Transfer Functions (CTFs). Initially, the Weighted Prediction Error (WPE) algorithm and the Delay and Sum (DAS) beamformer are employed to acquire an initial estimate of the target source signal at the source's location. Subsequently, the CTF coefficients are calculated employing either the Wiener filter or the Kalman filter, with their respective parameters optimized by the application of Particle Swarm Optimization. (PSO). To retrieve the impulse responses of ATFs, the short-time Fourier transform (STFT) of a unit pulse sequence is convolved with the CTF coefficients, followed by the application of the inverse STFT. To illustrate the efficacy of our proposed ATF estimation technique, we take the application of dereverberation using the Multiple Input/Output Inverse Theorem (MINT) as an example. This application necessitates precise ATF estimates. Simulation results, conducted using a linear array comprising 30 elements, demonstrate that our method produces ATF estimates that closely align with the true room impulse responses. Moreover, the MINT-dereverberated signals exhibit a remarkable resemblance to the dry

source signals, as indicated by two objective performance metrics.

Index Terms — convolutive transfer functions, weighted prediction error, delay and sum beamformer, Wiener filter, Kalman filter, particle swarm optimization, multiple input/output inverse theorem

CONTENTS

LIST OF ALGORITHMS

LIST OF FIGURES

LIST OF TABLES

Chapter 1. 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 paper outlines a methodology for the estimation of blind ATF through CTF approximation. To start the process, dereverberation and source signal extraction are needed to compute CTF coefficients with techniques like Weighted Prediction Error (WPE) [11] and Delay and Sum (DAS) beamforming [12] . Afterward, CTF coefficients are computed using Wiener filters [13] or adaptive filters, such as Recursive Least Squares (RLS) [14] and Kalman filter [15] . The parameters of previous mentioned filters are optimized using Particle Swarm Optimization (PSO) [16] and its enhanced version [17] . To acquire ATFs in the time domain, namely Room Impulse Responses (RIRs), from the CTF coefficients, the estimated CTF coefficients are convolved with the STFT of a time-shifted unit pulse sequence. Subsequently, the

resulting convolved sequence is subjected to processing using the inverse STFT.

The validation of the proposed ATF estimation method consists of two parts. One is to compute the matching error (ME) of the ground truth ATF and the estimated ATF. The other is to compare two versions of the processed source signal. One version of the signal is dereverberated using the Multiple Input/Output Inverse Theorem (MINT) [18] together with the estimated RIRs, while the other variant is dereverberated using the state-of-the-art WPE method, following DAS beamformer. The efficiency of the processed source signal is assessed based on several metrics, including the Perceptual Evaluation of Speech Quality (PESQ) [19] and Signal-to-Distortion Ratio (SDR) [20] . The simulations encompass diverse reverberation times and employ a Uniform Linear Array (ULA) of thirty microphones. The results demonstrate that the proposed approach outperforms WPE following DAS beamformer in various reverberant environments.

Chapter 2. CTF SIGNAL MODEL

2.1. 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 [21] and [22] . 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 is given by

$$x_{p,k} = \sum_{m} x(m)\tilde{w}^{*}_{p,k}(m), \tag{1}$$

where

$$\tilde{w}_{p,k}(n) \triangleq \tilde{w}(n - pD)e^{j\frac{2\pi}{N}k(n-pD)}. \tag{2}$$

An analysis window of length N is denoted by $\tilde{w}(n)$. The frame index is denoted by $p \in [1, P]$, and $k \in [0, N–1]$ represents the frequency-band index. The discrete-time shift is denoted by D. Complex conjugation is represented by *. The reconstruction of $x(n)$ which is inverse STFT is achieved by

$$x(n) \triangleq \sum_{p}\sum_{k=0}^{N-1} x_{p,k} w_{p,k}(n), \tag{3}$$

where

$$w_{p,k}(n) \triangleq w(n-pD)e^{j\frac{2\pi}{N}k(n-pD)}, \quad (4)$$

and $w(n)$ denotes a synthesis window of length N. This paper assumes $\tilde{w}(n)$ and $w(n)$ are real functions. By substituting (1) into (3), we acquire the completeness condition

$$\sum_{p} w(n-pD)\tilde{w}(n-pD) = \frac{1}{N} \quad \text{for all } n. \quad (5)$$

If the analysis and synthesis windows meet the requirements outlined in (5), the signal $x(n)$ can be reconstructed flawlessly using its STFT coefficients $x_{p,k}$. However, for $D \leqq N$ and for a given synthesis window $w(n)$, there might be an infinite number of solutions to (5); thus, the choice of the analysis window may not be unique [23] , [24] .

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:

$$o(n) = \sum_{i=0}^{Q-1} h(i)x(n-i). \quad (6)$$

From (1) and (6), the STFT of $o(n)$ can be written as

$$o_{p,k} = \sum_{m,l} h(l)x(m-l)\tilde{w}^{*}_{p,k}(m). \quad (7)$$

Substituting (3) into (7), we obtain

$$\begin{aligned} o_{p,k} &= \sum_{m,l} h(l) \sum_{k'=0}^{N-1} \sum_{p'} x_{p',k'} w_{p',k'}(m-l)\tilde{w}^{*}_{p,k}(m) \\ &= \sum_{k'=0}^{N-1} \sum_{p'} x_{p',k'} h_{p,k,p',k'} \end{aligned}, \quad (8)$$

where

$$h_{p,k,p',k'} = \sum_{m,l} h(l) w_{p',k'}(m-l) \tilde{w}^*_{p,k}(m) \qquad (9)$$

may be interpreted as the STFT of $h(n)$ using a composite analysis window $\sum_m w_{p',k'}(m-l)\, \tilde{w}_{p,k}^*(m)$. Substituting (2) and (4) into (9), we obtain

$$\begin{aligned} h_{p,k,p',k'} &= \sum_{m,l} h(l) w(m-l-p'D) e^{j\frac{2\pi}{N}k'(m-l-p'D)} \times \tilde{w}(m-pD) e^{-j\frac{2\pi}{N}k(m-pD)} \\ &= \sum_{l} h(l) \sum_{m} \tilde{w}(m) e^{-j\frac{2\pi}{N}km} \times w((p-p')D-l+m) e^{j\frac{2\pi}{N}k'((p-p')D-l+m)}, \\ &= \left\{ h(n) * \zeta_{k,k'}(n) \right\}\Big|_{n=(p-p')D} \triangleq h_{p-p',k,k'} \end{aligned} \qquad (10)$$

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

$$\zeta_{k,k'}(n) = e^{-j\frac{2\pi}{N}k'n} \sum_{m} \tilde{w}(m) w(n+m) e^{-j\frac{2\pi}{N}m(k-k')}. \qquad (11)$$

From (10), we know that $h_{p,k,p',k'}$ depends on $(p - p')$ rather than on p and p' separately. Substituting (10) into (8), we obtain

$$o_{p,k} = \sum_{k'=0}^{N-1} \sum_{p'} x_{p',k'} h_{p-p',k,k'} = \sum_{k'=0}^{N-1} \sum_{p'} x_{p-p',k'} h_{p',k,k'}. \qquad (12)$$

From (10), we also obtain

$$h_{p,k,k'} = \left\{ h(n) * \zeta_{k,k'}(n) \right\}\Big|_{n=pD}. \qquad (13)$$

From (11), we get

$$\zeta_{k,k'}(n) = e^{-j\frac{2\pi}{N}k'n} \sum_{m} \tilde{w}(m) w(n+m) e^{-j\frac{2\pi}{N}m(k-k')} = e^{-j\frac{2\pi}{N}k'n} w_{n,k-k'}, \qquad (14)$$

where $w_{n,k}$ is the STFT representation of the synthesis window $w(n)$ calculated with a decimation factor L=1. Equation (12) demonstrates that for a particular frequency-band index k, the temporal signal can be acquired by convolving the signal $x_{p,k'}$ in each frequency-band k'(k'= 0,1,…,N-1) with its corresponding filter $h_{p,k,k'}$ and subsequently

adding up all the outputs. Here, the term for $k = k'$ is referred to as a band-to-band filter, and $k \neq k'$ is referred to as a crossband filter, and crossband filters are employed to eliminate the aliasing effects resulting from subsampling. Note that (13) indicates that, in general, for fixed k and k', the filter $h_{p,k,k'}$ has $\lceil N/D \rceil - 1$ non-causal coefficients. Hence, in echo cancellation applications, these coefficients must be taken into consideration. Extra delay of $(\lceil N/D \rceil - 1)\ D$ samples is typically introduced into the microphone signal to deal with this problem, as illustrated in [25] .

2.2. 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

$$y(n) = a(n) * s(n), \qquad (15)$$

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 (15) is often estimated using MTF in the STFT domain, as demonstrated by

$$y_{p,k} = a_k s_{p,k}, \qquad (16)$$

where $y_{p,k}$ and $s_{p,k}$ represent the STFTs of their respective signals, while a_k denotes the

Fourier transform of the RIR $a(n)$. In addition, $p \in [1, P]$ refers to the frame index, N indicates the STFT window size, and $k \in [0, N-1]$ represents the frequency index as in crossband filter. However, it is important to note that the approximation in (16) 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 (12) the STFT coefficient $y_{p,k}$ is expressed as the sum of several convolutions between the STFT-domain source signal and the filter across the frequency bins, as follows:

$$y_{p,k} = \sum_{k'=0}^{N-1} \sum_{p'} s_{p-p',k'} \, a_{p',k,k'} \,. \qquad (17)$$

Assuming D is the STFT frame step as stated above, if D is less than N, then $a_{p',k,k'}$ becomes non-causal, with $\lceil N/D \rceil - 1$ 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' 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 $\lceil N/D \rceil - 1$ of the frame index for the received microphone signal [25] . From (13) the STFT domain impulse response $a_{p',k,k'}$ relates to the time domain impulse response $a(n)$ by

$$a_{p',k,k'} = (a(n) * \zeta_{k,k'}(n))\big|_{n=p'D}, \quad (18)$$

which indicates the convolution with respect to the time index n evaluated at frame steps using (14). Note that for the remainder of this article, we will continue to refer to the analysis and synthesis windows in the STFT procedure as $\tilde{w}(n)$ 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 $k = k'$, as described in the following:

$$y_{p,k} \approx \sum_{p'=0}^{L-1} s_{p-p',k}\, a_{p',k} = s_{p,k} * a_{p,k}. \quad (19)$$

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

$$y_{p,k}^{i} = \sum_{p'=0}^{L-1} s_{p-p',k}\, a_{p',k}^{i}, \quad (20)$$

where $y_{p,k}^{i}$ and $a_{p,k}^{i}$ represent the i-th microphone signal and the corresponding CTF, respectively. Therefore, the source signals can be expressed in matrix form as follow:

$$\underbrace{\begin{bmatrix} y_{p,k}^{1} \\ y_{p,k}^{2} \\ \vdots \\ y_{p,k}^{M} \end{bmatrix}}_{\mathbf{y}_d} = \underbrace{\begin{bmatrix} a_{0,k}^{1} & a_{1,k}^{1} & \cdots & a_{L-1,k}^{1} \\ a_{0,k}^{2} & a_{1,k}^{2} & \cdots & a_{L-1,k}^{2} \\ \vdots & \vdots & \ddots & \vdots \\ a_{0,k}^{M} & a_{1,k}^{M} & \cdots & a_{L-1,k}^{M} \end{bmatrix}}_{\mathbf{A}} \underbrace{\begin{bmatrix} s_{p,k} \\ s_{p-1,k} \\ \vdots \\ s_{p-(L-1),k} \end{bmatrix}}_{\mathbf{s}}. \quad (21)$$

Since the proposed algorithm functions on a frequency basis, the frequency index will be omitted henceforth for the sake of brevity. From (21) we can rewrite the matrix form with respect to frame index as follow:

$$\mathbf{y}_{d,p} = \mathbf{A} \cdot \mathbf{s}_p, \quad (22)$$

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 (22).

Chapter 3. PROPOSED METHOD

In this section, we introduce a technique for estimating the ATF in a blind manner. It should be emphasized that we possess solely the positions of the microphones and the source, alongside the delayed microphone signal $\mathbf{y}_{d,p}$ (by ⌈N/D⌉ – 1 frames [25]) and pre-processed source signal acquired via the non-delayed microphone signal $\mathbf{y}_p$, which will undergo processing with the WPE and DAS beamformer. Notably, we provide three techniques for estimating the CTF coefficients: the Wiener filter, RLS and Kalman adaptive filter.

3.1. Pre-processing

In order for the subsequent CTF estimation algorithm to function effectively, it is necessary to have access to clean and echo-free source signal. However, in practical applications, obtaining clean source signal is often challenging. Therefore, in this paper, we employ the Weighted Prediction Error (WPE) method for dereverberation [11] . Subsequently, we utilize the Delay and Sum (DAS) beamformer [12] to extract clean source signal from the WPE outputs of all channels.

3.1.1. WPE

If only one speech source is captured by M microphones, then we may rewrite (15) as follows:

$$y^m(n) = \sum_{k=0}^{L_a-1} a^m(k)s(n-k)\,, \qquad (23)$$

where m and L_a represent the ordinal numbers of the *m-th* microphone and length of the RIR. The reverberant signal $y^m(n)$ in (23) consists of three parts: a direct signal, early reverberation, and late reverberation. It is common practice to take the first two components as the desired signal, which is denoted by $d^m(n)$. Meanwhile, the late reverberation is taken as the signal to be eliminated and is denoted by $r^m(n)$. The relationship between these signals can be expressed as such:

$$y^m(n) = d^m(n) + r^m(n)\,, \qquad (24)$$

where

$$\begin{aligned} d^m(n) &= \sum_{k=0}^{C-1} a^m(k)s(n-k) \\ r^m(n) &= \sum_{k=C}^{L_a-1} a^m(k)s(n-k)\,, \end{aligned} \qquad (25)$$

where C 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 $a^m(n)$ in different channels do not share common zeros, the relationship between speech signal and the microphone signals in (23) can be rewritten (stepwise derivation is shown in [26]) as

$$y^m(n) = (\mathbf{c}^m)^T \mathbf{y}(n-C) + d^m(n)\,, \qquad (26)$$

where T denotes matrix transpose and

$$\mathbf{y}(n) = \left[(\mathbf{y}^1(n))^T, (\mathbf{y}^2(n))^T \right]^T \in \mathbb{R}^{2L_c \times 1}$$

$$\mathbf{y}^m(n) = \left[y^m(n), y^m(n-1), \cdots, y^m(n-L_c+1) \right]^T \in \mathbb{R}^{L_c \times 1}$$

$$\mathbf{c}^m = \mathbf{H}(\mathbf{H}^T\mathbf{H})^{-1}\mathbf{a}_{late}^m \in \mathbb{R}^{2L_c \times 1}$$

$$\mathbf{H} = \begin{bmatrix} a^1(0) & a^1(1) & \cdots & a^1(L_a-1) & 0 & \cdots & \cdots & 0 \\ 0 & a^1(0) & a^1(1) & \cdots & a^1(L_a-1) & 0 & \cdots & 0 \\ & & & \ddots & & & & \\ 0 & \cdots & \cdots & 0 & a^1(0) & a^1(1) & \cdots & a^1(L_a-1) \\ a^2(0) & a^2(1) & \cdots & a^2(L_a-1) & 0 & \cdots & \cdots & 0 \\ 0 & a^2(0) & a^2(1) & \cdots & a^2(L_a-1) & 0 & \cdots & 0 \\ & & & \ddots & & & & \\ 0 & \cdots & \cdots & 0 & a^2(0) & a^2(1) & \cdots & a^2(L_a-1) \end{bmatrix} \in \mathbb{R}^{2L_c \times L_H} \quad (27)$$

$$\mathbf{a}_{late}^m = \left[a^m(C), a^m(C+1), \cdots, a^m(L_a-1), 0, \cdots, 0 \right] \in \mathbb{R}^{L_H \times 1},$$

where $\mathbf{c}^m$ and L_c indicate the vector of regression coefficients and the regression order, respectively, and L_H equals $L_c + L_a - 1$.

Using the estimated vector of regression coefficients $\hat{\mathbf{c}}^m$ and following (26), it is possible to acquire the desired signal as follow:

$$\hat{d}^m(n) = y^m(n) - (\hat{\mathbf{c}}^m)^T \mathbf{y}(n-C). \quad (28)$$

Hence, the dereverberation can be achieved by obtaining a suitable estimated vector of regression coefficients $\hat{\mathbf{c}}^m$ from the microphone signals. Because $\hat{\mathbf{c}}^1$ is completely determined independently of $\hat{\mathbf{c}}^2$, in the following, we disregard the optimization of $\hat{\mathbf{c}}^2$ without loss of generality. The resultant optimization algorithm can be summarized (stepwise derivation is shown in [11]) as follow.

Algorithm 1 WPE

Input: $y^1(n)$, $\mathbf{y}(n)$

1) Initialize $\hat{\sigma}(n)^2$ as

$$\hat{\sigma}(n)^2 = \max\left\{\frac{1}{L_f}\sum_{n'=n-\frac{L_f}{2}+1}^{n+\frac{L_f}{2}+1}\left|y^1(n')\right|^2,\ \mu\right\}, \tag{29}$$

where L_f is length of short time frame and $\mu > 0$ is a certain lower bound to avoid zero division.

2) Repeat the following steps until convergence.

a. Update $\hat{\mathbf{c}}^1$ as follows:

$$\hat{\mathbf{c}}^1 = \hat{\mathbf{\Phi}}^+ \hat{\mathbf{\Lambda}}, \tag{30}$$

where $^+$ denotes the pseudo inverse and

$$\hat{\mathbf{\Phi}} = \sum_{n=1}^{\tau} \frac{\mathbf{y}(n-C)\mathbf{y}(n-C)^T}{\hat{\sigma}(n)^2} \tag{31}$$

$$\hat{\mathbf{\Lambda}} = \sum_{n=1}^{\tau} \frac{\mathbf{y}(n-C)y^1(n)}{\hat{\sigma}(n)^2}, \tag{32}$$

where τ is the largest sample index of the microphone signal.

b. Update $\hat{d}^1(n)$ as $\hat{d}^1(n) = y^1(n) - (\hat{\mathbf{c}}^1)^T \mathbf{y}(n-D)$.

c. Update $\hat{\sigma}(n)^2$ as follow:

$$\hat{\sigma}(n)^2 = \max\left\{\frac{1}{L_f}\sum_{n'=n-\frac{L_f}{2}+1}^{n+\frac{L_f}{2}+1}\left|\hat{d}^1(n)\right|^2,\ \mu\right\}. \tag{33}$$

It is worth noting that we will refer to the WPE output of the *m-th* channel $\hat{d}^m(n)$

as $y_{WPE}^{m}(n)$ from this point forward.

3.1.2. DAS Beamformer

Once the dereverberated microphone signal $y_{WPE}^{m}(n)$ from WPE is obtained, clean source signal extraction through DAS beamformer can be executed. First, we convert $y_{WPE}^{m}(n)$ to the STFT domain, resulting in $y_{WPE,p}^{m}$. Here, p still represents the frame index as previously explained. Secondly, we calculate the beamforming weight of the DAS beamformer as follow:

$$\mathbf{w}_{DAS} = \frac{1}{M}\left[e^{\frac{-j\kappa R^1}{R^1}}, e^{\frac{-j\kappa R^2}{R^2}}, \cdots, e^{\frac{-j\kappa R^M}{R^M}} \right]^T, \quad (34)$$

where κ and R^m denote wave number and distance between source and *m-th* microphone, respectively. Finally, we perform the inner product between the weights and $y_{WPE,p}^{m}$ to get clean source signal as follow:

$$s_{DAS,p} = \mathbf{w}_{DAS}^{H} \begin{bmatrix} y_{WPE,p}^{1} \\ y_{WPE,p}^{2} \\ \vdots \\ y_{WPE,p}^{M} \end{bmatrix}, \quad (35)$$

where H denotes Hermitian transpose.

3.2. Wiener Filtering Approach

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

$$\min_{\mathbf{A}\in\mathbb{C}^{M\times L}} E\left[\left\| \mathbf{y}_{d,p} - \hat{\mathbf{A}}\mathbf{s}_{DAS,p} \right\|_2^2 \right], \quad (36)$$

where E[·] denotes the expectation with respect to the frames. Therefore, (36) can be rewritten as

$$\min_{\mathbf{A}\in\mathbb{C}^{M\times L}} tr\left\{\mathbf{R}_{\mathbf{yy}} - \hat{\mathbf{A}}\mathbf{R}_{\mathbf{sy}} - \mathbf{R}_{\mathbf{sy}}^{H}\hat{\mathbf{A}}^{H} + \hat{\mathbf{A}}\mathbf{R}_{\mathbf{ss}}\hat{\mathbf{A}}^{H}\right\}, \tag{37}$$

where $tr\{\cdot\}$ denotes the matrix trace, and the associated covariance matrices is

$$\begin{aligned}\mathbf{R}_{\mathbf{yy}} &= E\left[\mathbf{y}_{d,p}\mathbf{y}_{d,p}^{H}\right] \in \mathbb{C}^{M\times M}\\ \mathbf{R}_{\mathbf{ss}} &= E\left[\mathbf{s}_{DAS,p}\mathbf{s}_{DAS,p}^{H}\right] \in \mathbb{C}^{L\times L}\\ \mathbf{R}_{\mathbf{sy}} &= E\left[\mathbf{s}_{DAS,p}\mathbf{y}_{d,p}^{H}\right] \in \mathbb{C}^{L\times M}\end{aligned}. \tag{38}$$

By taking the derivative of (37) with respect to $\hat{\mathbf{A}}^{H}$, we obtain

$$\nabla_{\hat{\mathbf{A}}^{H}} J = -2\mathbf{R}_{\mathbf{sy}} + 2\mathbf{R}_{\mathbf{ss}}\hat{\mathbf{A}}^{H} = 0. \tag{39}$$

The optimal Wiener solution can be obtained as

$$\hat{\mathbf{A}} = \mathbf{R}_{\mathbf{sy}}^{H}\mathbf{R}_{\mathbf{ss}}^{-1}. \tag{40}$$

In practical implementation, instead of the expectation, the recursive averaging is adopted to obtain $\mathbf{R}_{\mathbf{sy}}$ and $\mathbf{R}_{\mathbf{ss}}$ as given by

$$\begin{aligned}\mathbf{R}_{\mathbf{ss},p} &= \alpha\mathbf{R}_{\mathbf{ss},p} + (1-\alpha)\mathbf{s}_{DAS,p}\mathbf{s}_{DAS,p}^{H},\\ \mathbf{R}_{\mathbf{sy},p} &= \alpha\mathbf{R}_{\mathbf{sy},p} + (1-\alpha)\mathbf{s}_{DAS,p}\mathbf{y}_{d,p}^{H}\end{aligned} \tag{41}$$

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

Algorithm 2 CTF estimation using Wiener filtering

Input: $\mathbf{y}_{d,p}$, $\mathbf{s}_{DAS,p}$

1) Initialize forgetting factor α and covariance matrices as $\mathbf{R}_{\mathbf{ss},0} = \mathbf{0} \in \mathbb{C}^{L\times L}$, $\mathbf{R}_{\mathbf{sy},0} = \mathbf{0} \in \mathbb{C}^{L\times M}$

---

2) For each instant of frame, $p = 1, 2, \ldots$, compute

$$\begin{aligned}\mathbf{R}_{\mathbf{ss},p} &= \alpha\mathbf{R}_{\mathbf{ss},p} + (1-\alpha)\mathbf{s}_{DAS,p}\mathbf{s}_{DAS,p}^{H} \\ \mathbf{R}_{\mathbf{sy},p} &= \alpha\mathbf{R}_{\mathbf{sy},p} + (1-\alpha)\mathbf{s}_{DAS,p}\mathbf{y}_{d,p}^{H} \\ \hat{\mathbf{A}}_{p} &= \mathbf{R}_{\mathbf{sy},p}^{H}\mathbf{R}_{\mathbf{ss},p}^{-1}\end{aligned} \quad (42)$$

It should be noted that the estimated matrix of CTF coefficients $\widehat{\mathbf{A}}$ changes depending on the processing frame, and the accuracy improves with an increase in the number of processed frames.

3.3. 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 [14] is currently being conducted in the complex domain. The RLS algorithm aims to minimize the sum of the weighted error norm square as

$$\min_{\mathbf{A}\in\mathbb{C}^{M\times L}} \sum_{i=1}^{p} \lambda^{p-i} \left\| \mathbf{y}_{d,i} - \hat{\mathbf{A}}_{i}\mathbf{s}_{DAS,i} \right\|_{2}^{2}, \quad (43)$$

where p represents both the adaptation iteration and the frame index, while λ represents the forgetting factor, which imparts weight to the square of the error norm concerning the iteration. Guided by the objective function articulated in (43), the RLS algorithm is employed for the estimation of the CTF coefficients matrix $\widehat{\mathbf{A}}$. The RLS approach can be succinctly summarized in the subsequent algorithmic routine.

Algorithm 3 CTF estimation using RLS

Input: $\mathbf{y}_{d,p}$, $\mathbf{s}_{DAS,p}$

1) Initialize RLS forgetting factor λ, weight and inverse of correlation matrix as

$\mathbf{w}_{RLS,0} = \mathbf{0} \in \mathbb{C}^{L\times M}$, $\mathrm{P}_{RLS,0} = \varepsilon^{-1}\mathbf{I} \in \mathbb{C}^{L\times L}$, where ε is a small positive constant

2) For each instant of frame, p = 1, 2, …, compute

$$\begin{aligned}
&\mathbf{e}_p = \mathbf{y}_{d,p}^H - \mathbf{s}_{DAS,p}\mathbf{w}_{RLS,p-1} \\
&\mathbf{k}_p = \frac{\lambda^{-1}\mathbf{P}_{RLS,p-1}\mathbf{s}_{DAS,p}}{1+\lambda^{-1}\mathbf{s}_{DAS,p}^H\mathbf{P}_{RLS,p-1}\mathbf{s}_{DAS,p}} \\
&\mathbf{w}_{RLS,p} = \mathbf{w}_{RLS,p-1} + \mathbf{k}_p\mathbf{e}_p \\
&\mathbf{P}_{RLS,p} = \lambda^{-1}\mathbf{P}_{RLS,p-1} - \lambda^{-1}\mathbf{k}_p\mathbf{s}_{DAS,p}^H\mathbf{P}_{RLS,p-1} \\
&\hat{\mathbf{A}}_p = \mathbf{w}_{RLS,p}^H
\end{aligned} \quad . \qquad (44)$$

It should be noted that the estimated matrix of CTF coefficients $\hat{\mathbf{A}}$ changes depending on the processing frame too. It should be noted that the estimated matrix of CTF coefficients $\hat{\mathbf{A}}$ changes depending on the processing frame, and the accuracy improves with an increase in the number of processed frames too.

3.4. Kalman Algorithm Adaptive Filtering Approach

For the third technique, the CTF coefficients matrix is estimated by applying the Kalman filter. It is worth noting that in this paper, 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 unchanged. The process equation of stationary Kalman adaptive filter of each microphone without process noise is described as

$$\mathbf{w}_{Kalman,p}^{m\ *} = \mathbf{w}_{Kalman,p-1}^{m\ *}, \qquad (45)$$

where $\mathbf{w}_{Kalman,p}^{m} \in \mathbb{C}^{Lx1}$ signifies the optimal weight vector and has a connection with the CTF coefficients matrix as

$$\mathbf{A}_{p}^{m} = \mathbf{w}_{Kalman,p}^{m\ H}, \quad (46)$$

where $\mathbf{A}_{p}^{m}$ denotes the *m-th* row of **A**. The measurement equation of stationary Kalman adaptive filter of each microphone is described as

$$y_{d,p}^{m} = \mathbf{s}_{DAS,p}^{T} \mathbf{w}_{Kalman,p}^{m\ *} + e_{p}^{m}, \quad (47)$$

where e_{p}^{m} denotes the measurement noise for each microphone, and

$$E\left[e_{p}^{m} e_{p}^{m*}\right] = R_{p}^{m}, \quad (48)$$

where R_{p}^{m} is the covariance of measurement noise.

Using the process and measurement equations outlined in (45) and (47), the Kalman gain can be derived by minimizing the error covariance matrix [15] . 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: $y_{d,p}^{m}$, $\mathbf{s}_{DAS,p}$

1) Initialize estimated Kalman weight, error covariance matrix, Kalman gain and measurement noise covariance matrix as

   $\mathbf{w}_{Kalman,0}^{m} = \mathbf{0} \in \mathbb{C}^{L\times 1}$, $\mathbf{P}_{Kalman,0}^{m} = \eta\mathbf{I} \in \mathbb{C}^{L\times L}$, $\mathbf{K}_{0}^{m} = \mathbf{0} \in \mathbb{C}^{L\times 1}$, $\mathrm{R}^{m} = \rho \in \mathbb{C}^{1\times 1}$, where η and ρ is a small positive constant

---

2) For each microphone, $m = 1, 2, \ldots,$

For each instant of frame, $p = 1, 2, \ldots,$ compute

$$\begin{aligned}
&\mathbf{K}_p^m = \mathbf{P}_{Kalman,p}^m \mathbf{s}_{DAS,p} (\mathbf{s}_{DAS,p}{}^H \mathbf{P}_{Kalman,p-1}^m \mathbf{s}_{DAS,p} + \mathrm{R}^m)^{-1} \\
&\mathbf{w}_{Kalman,p}^m = \mathbf{w}_{Kalman,p-1}^m + \mathbf{K}_p^m (y_{d,p}^{m\ *} - \mathbf{s}_{DAS,p}{}^H \mathbf{w}_{Kalman,p-1}^m) \cdot \\
&\mathbf{P}_{Kalman,p}^m = \mathbf{P}_{Kalman,p-1}^m - \mathbf{K}_p^m \mathbf{s}_{DAS,p}{}^H \mathbf{P}_{Kalman,p-1}^m \\
&\hat{\mathbf{A}}_p^m = \mathbf{w}_{Kalman,p}^{m\ H}
\end{aligned} \tag{49}$$

It is easy to observe that the estimated matrix of CTF coefficients $\widehat{\mathbf{A}}$ exhibits variability contingent upon the processing frame, and the accuracy improves with an increase in the number of processed frames too.

3.5. ATF Reconstruction

Once the matrix of CTF coefficients $\widehat{\mathbf{A}}$ 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 $\delta(n)$, which experiences a delay of $(L-1)D$ points as

$$\delta(n) = \begin{cases} 1, & n = (L-1)D \\ 0, & \text{else } n \end{cases}. \tag{50}$$

Subsequently, it is transformed into the STFT domain, resulting in δ_p. Ultimately, the estimated CTF coefficients are convolved with the transformed unit pulse sequence δ_p to yield the following signal:

$$\hat{G}_p^m = \sum_{p'=0}^{L-1} \delta_{p+L-p'}\ \hat{a}_{p'}^m, \tag{51}$$

where $p \in [0, P_{ATF}]$. The estimated RIRs $\hat{g}^m(n)$ $n \in [0, N_{ATF}]$ can be obtained by

applying the inverse STFT to $\hat{G}_p^m$. Subsequently, the estimated ATFs, represented by vector $\hat{\mathbf{g}}^m$ with each element corresponding to different frequency bins, can be obtained by performing a fast Fourier transform (FFT) on the estimated RIRs $\hat{g}^m(n)$. $\hat{\mathbf{g}}^m$ is expressed as

$$\hat{\mathbf{g}}^m = \left[\hat{g}_1^m, \hat{g}_2^m, \cdots, \hat{g}_{N_{ATF}}^m \right]^T . \quad (52)$$

3.6. 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 [17] to optimize the parameters that are involved.

3.6.1 PSO

The PSO algorithm is a swarm intelligent optimization technique inspired by the flocking of birds and schooling of fish [16] . 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:

$$\begin{aligned} \mathbf{X}_j(t) &= \left[x_j^1(t) \quad x_j^2(t) \quad \cdots \quad x_j^U(t) \right] \\ \mathbf{V}_j(t) &= \left[v_j^1(t) \quad v_j^2(t) \quad \cdots \quad v_j^U(t) \right] \end{aligned} . \quad (53)$$

Let the fitness function $f:\mathbb{R}^U \to \mathbb{R}$ 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:

$$f(\mathbf{X}_j(t)) = \sum_p \left\| \mathbf{y}_{d,p} - \hat{\mathbf{A}}(\mathbf{X}_j(t))\mathbf{s}_{DAS,p} \right\|_2 , \quad (54)$$

where $\widehat{\mathbf{A}}(\boldsymbol{X}_J(t))$ represents the estimated CTF coefficients matrix obtained from any one of the three methods when the parameters $\boldsymbol{X}_J(t)$ are specified. After calculating the fitness value of the entire population, $pbest_j(t)$ 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.

$$\begin{aligned} pbest_j(t) &= \left[pbest_j^1(t), pbest_j^2(t), \cdots, pbest_j^U(t) \right] \\ gbest(t) &= \left[gbest^1(t), gbest^2(t), \cdots, gbest^U(t) \right] \end{aligned} \quad (55)$$

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

$$\begin{aligned} \mathbf{V}_j(t+1) &= w_{in} * \mathbf{V}_j(t) + c_1 * r_1 * (pbest_j(t) - \mathbf{X}_j(t)) + c_2 * r_2 * (gbest(t) - \mathbf{X}_j(t)) \\ \mathbf{X}_j(t+1) &= \mathbf{X}_j(t) + \mathbf{V}_j(t+1) \end{aligned} , \quad (56)$$

where w_{in} 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 T_{max} has not been reached. The PSO algorithm's entire process is presented in Figure 1.

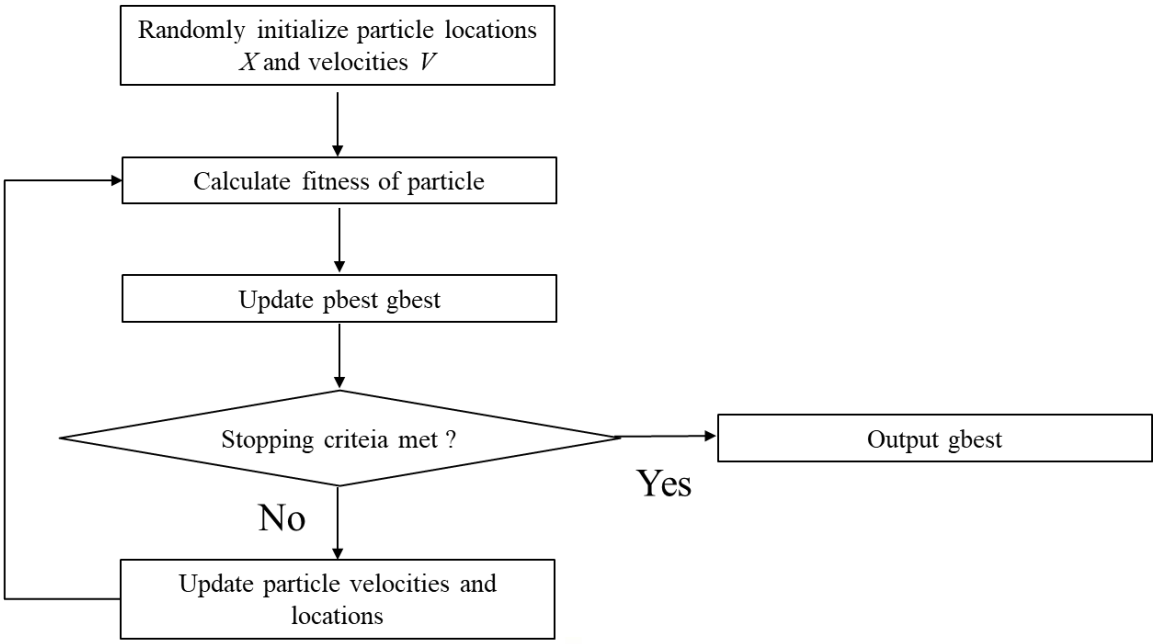


Figure 1 Block diagram of PSO

3.6.2 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 [17] . 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 w_{in} 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 [27] . The formula for calculating inertia weight is

$$\begin{aligned} z_t &= C_{in} \cdot z_{t-1} \cdot (1 - z_{t-1}), \quad z_t \in (0,\ 1) \\ w_{in}(t) &= (w_{\max} - w_{\min}) \cdot \frac{(T_{\max} - t)}{T_{\max}} + w_{\min} \cdot z_t, \end{aligned} \tag{57}$$

where C_{in}, w_{max}, w_{min} and T_{max} 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 ($pbest_j(t)$ 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, [17] 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 $pbest_j(t)$ 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 ($Fpbest_j(t)$). The learning strategy is expressed as

$$Cpbest(t) = \arg\min\left\{ f(pbest_a(t)), f(pbest_b(t)), \cdots, f(pbest_d(t)) \right\}, \quad a \neq b \neq c \neq d$$
$$Fbest_j(t) = \begin{cases} Cpbest(t), & f(Cpbest(t)) < f(pbest_j(t)) \\ pbest_j(t), & f(Cpbest(t)) \geq f(pbest_j(t)) \end{cases} . \quad (58)$$

An excessive focus on $gbest(t)$ can lead to a rapid diversity in population. To mitigate this potential problem, [17] 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

$$Mpbest(t) = \frac{1}{N}\left[\sum_{j=1}^{J} pbest_j^1(t), \sum_{j=1}^{J} pbest_j^2(t), \cdots, \sum_{j=1}^{J} pbest_j^U(t) \right]. \quad (59)$$

Finally, the velocity update equation is changed to:

$$V_j(t+1) = w_{in}(t)V_j(t+1) + c_1 \cdot r_1 \cdot (Fpbest_j(t) - X_j(t)) + c_2 \cdot r_2 \cdot (Mpbest(t) - X_j(t)) . \quad (60)$$

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 [28] . 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 [17] . This strategy is articulated by

$$\beta_j(t) = \frac{e\exp(f(X_j(t)))}{e\exp(\frac{1}{J}\sum_{j=1}^{J} f(X_j(t)))}$$
$$X_j(t+1) = \begin{cases} D_j \cdot \exp(b \cdot l) \cdot \cos(2\pi l) + gbest(t), & \beta_j(t) < r \\ X_j(t) + V_j(t+1), & \beta_j(t) \geq r \end{cases}, \quad (61)$$
$$D_j = \left\| gbest(t) - X_j(t) \right\|_2$$

where D_j 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 $\beta_j(t)$ is calculated by evaluating the fitness value of the current particle in relation to the average fitness value. If $\beta_j(t)$ 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 [17] . In each iteration, the worst-performing particle is identified and replaced, as defined by

$$
\begin{aligned}
&WX_j(t)=\text{argmax}\left\{ f(X_1(t)), f(X_2(t)),\cdots, f(X_J(t))\right\}\\
&NX_j(t) = gbest(t) + r_3 \cdot (pbest_e(t) - pbest_f(t)), \quad e \neq f \neq j \in [1,2,\cdots,J],\\
&WX_j(t) = \begin{cases} NX_j(t), & f(NX_j(t)) < f(WX_j(t)) \\ WX_j(t), & f(NX_j(t)) \geq f(WX_j(t)) \end{cases}
\end{aligned}
\tag{62}
$$

where $r_3 \in (0,1)$ 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:

$$
\begin{aligned}
&Nbest(t) = r_4 \cdot gbest(t) + (1-r_4) \cdot (gbest(t) - pbest_g(t)), \quad g \in [1,2,\cdots,J]\\
&gbest(t) = \begin{cases} Nbest(t), & f(Nbest(t)) < f(gbest(t)) \\ gbest(t), & f(Nbest(t)) \geq f(gbest(t)) \end{cases},
\end{aligned}
\tag{63}
$$

where $r_4 \in (0,1)$ is a random number.

The ASPSO algorithm's entire process is presented in Figure 2.

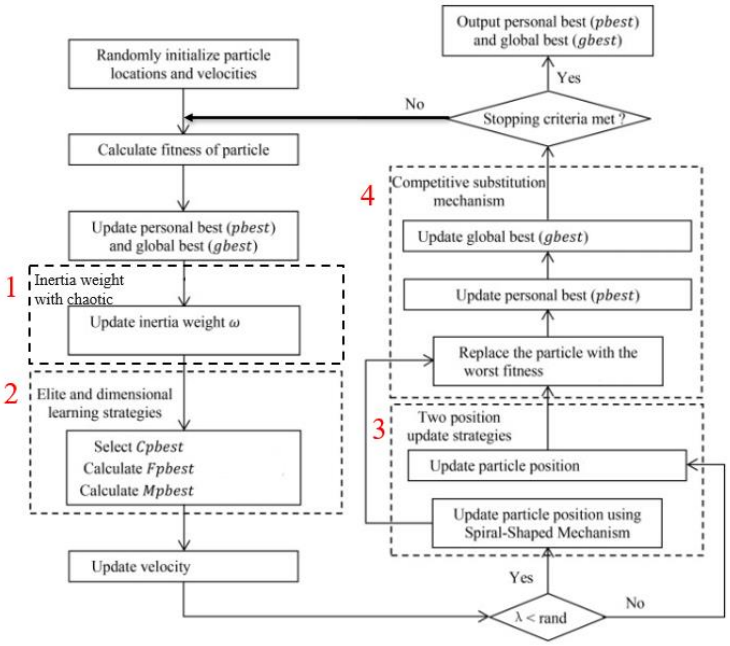


Figure 2 Block diagram of ASPSO

3.7. Summary of Proposed Method

Table 1 summarizes the method proposed in this paper, 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 $y^m(n)$ and delay it to get $y_d^m(n)$
Step 2: Do WPE followed by DAS to $y^m(n)$ obtaining $\mathbf{s}_{DAS,p}$
Step 3: Estimate the CTF coefficients matrix $\widehat{\mathbf{A}}$ via one of the algorithms below a. Algorithm 2 CTF estimation using Wiener filtering b. Algorithm 3 CTF estimation using RLS c. Algorithm 4 CTF estimation using stationary Kalman adaptive filtering

Step 4: Do (50) ~ (52) to obtain estimated RIRs $\hat{g}^m(n)$ or ATFs $\hat{\mathbf{g}}^m$

Step 5: Filter parameters optimization through PSO or ASPSO.

Step 6: Applications: MINT for dereverberation, etc.

Chapter 4. SIMULATIONS

4.1. Simulation Settings and Parameters

To evaluate the effectiveness of the proposed approach for estimating ATFs, we have employed MINT [18] to perform dereverberation using the estimated RIRs. In addition, we have also compared our approach with the state-of-the-art dereverberation method, WPE [11] , following DAS beamformer. The RIR dataset was generated using the RIR Generator [29] and the dimensions of the room were 5 m × 6 m × 2.5 m. The first sensor of a 30-microphone Uniform Linear Array (ULA) was situated at (1 m, 1.5 m, 1 m) with a 0.02 m gap along the x-axis. The room's layout is presented in Figure 3. At 16 kHz, speech signals were sampled and employed as sources for generating microphone signals, which were convolved with the RIRs of ground truth. The speech sources' location was at (2 m, 2.6 m, 1 m). The simulation verified seven diverse reverberation times, from T_{60} = 0.4s to 1.6s with an interval of 0.2s. A 1024-sample Hamming window (64 ms) with 75% overlap performed the STFT.

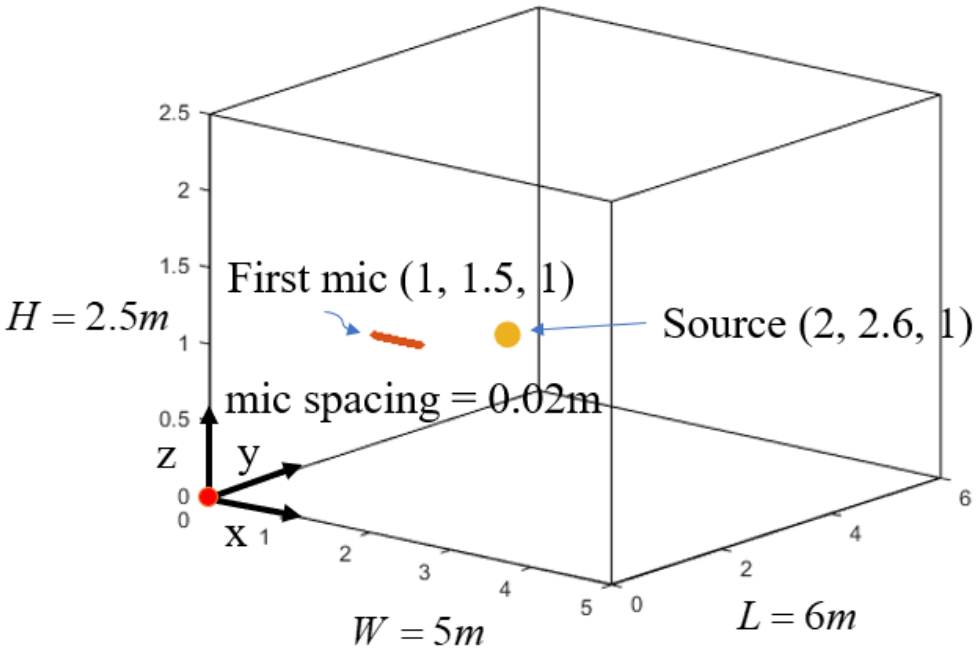


Figure 3 Configuration of the room for simulations

4.2. Results and Discussions

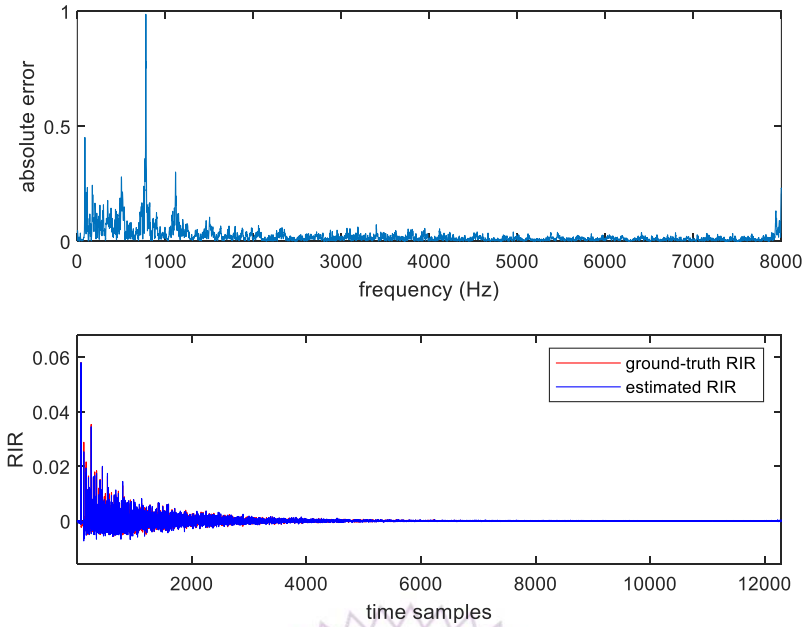
4.2.1 Without Parameters Optimization

Throughout the sections, 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 absolute error of the estimated ATF for all frequency bins and the magnitude of the estimated RIR compared to its ground truth values with T_{60} = 0.6 s for 10-*th* microphone and all algorithms are displayed in Figure 4. It is worth noting that in the case of blind estimation, it will inevitably encounter the equalization problem that there will be a scale gap between the estimated RIR and the

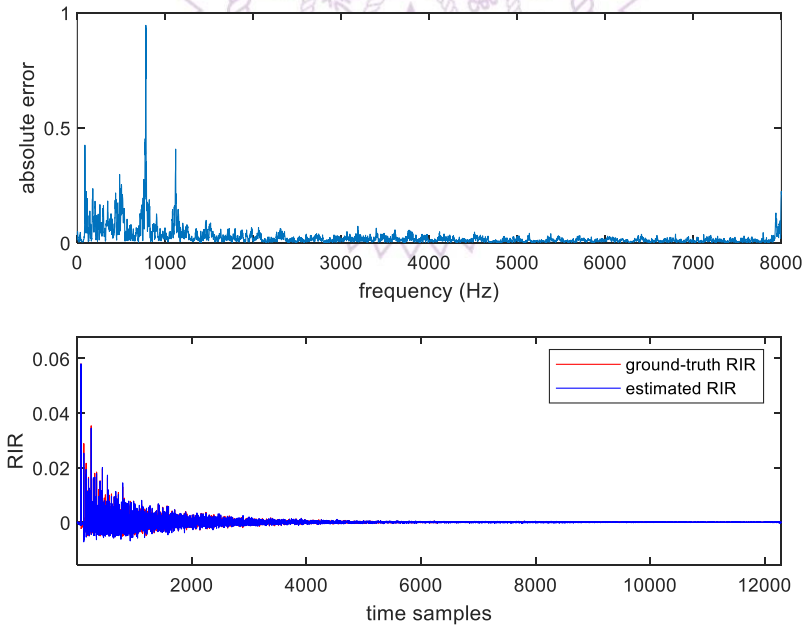
ground-truth RIR. To address this issue, we rescale the estimated RIR with the ratio calculated as the maximum absolute magnitude of the ground-truth RIR divided by the maximum absolute magnitude of the estimated RIR. Figure 4 reveals a minimal absolute error across all frequency bins and a remarkable correspondence between the magnitude of the estimated RIR and its ground-truth counterparts, thus fulfilling our requirements. Table 2 and Figure 5 demonstrate the ME of the estimated ATFs for all algorithms. The algorithms with the lowest ME at every reverberation time are highlighted in red. The ME is formulated as follows:

$$\mathrm{ME} = \frac{1}{M}\sum_{m=1}^{M}\left\|\mathbf{g}^{m} - \hat{\mathbf{g}}^{m}\right\|_{2}, \tag{64}$$

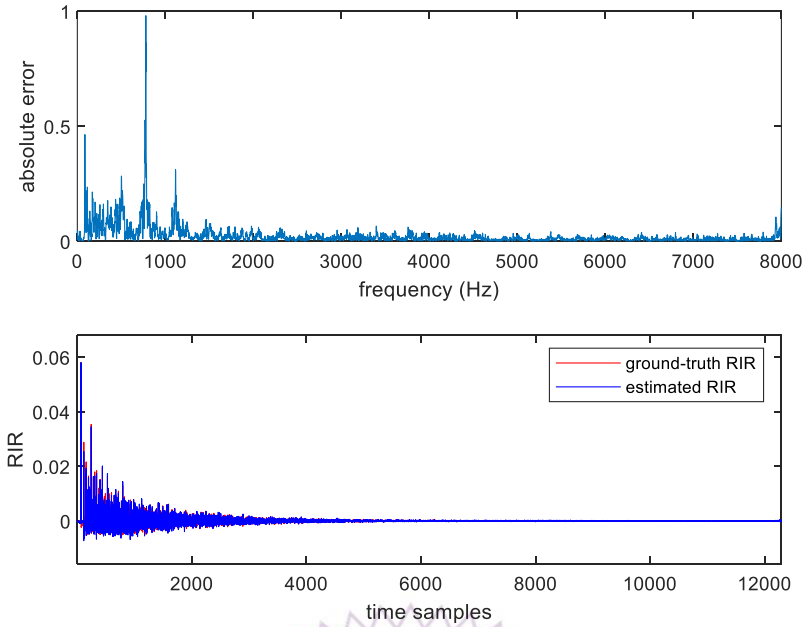
where $\mathbf{g}^{m}$ represents the ground-truth ATF of the *m-th* microphone. The results demonstrate a consistently low ME across all reverberation times, suggesting a favorable outcome.



(a) Wiener filter



(b) RLS



(c) Kalman filter

Figure 4 Absolute error of the estimated ATF and magnitude of the estimated RIR when T_{60} = 0.6s

Table 2 ME of the estimated ATFs based on the proposed methods in various reverberation times

T_{60} (s) / method	0.4	0.6	0.8	1	1.2	1.4	1.6
Wiener	2.8604	4.7624	6.058	7.6621	8.8369	10.2624	12.3786
RLS	2.9174	4.8753	6.9862	8.4617	9.3017	12.1247	14.6383
Kalman	2.8649	4.7647	5.9785	7.6486	8.9014	10.3768	12.4370

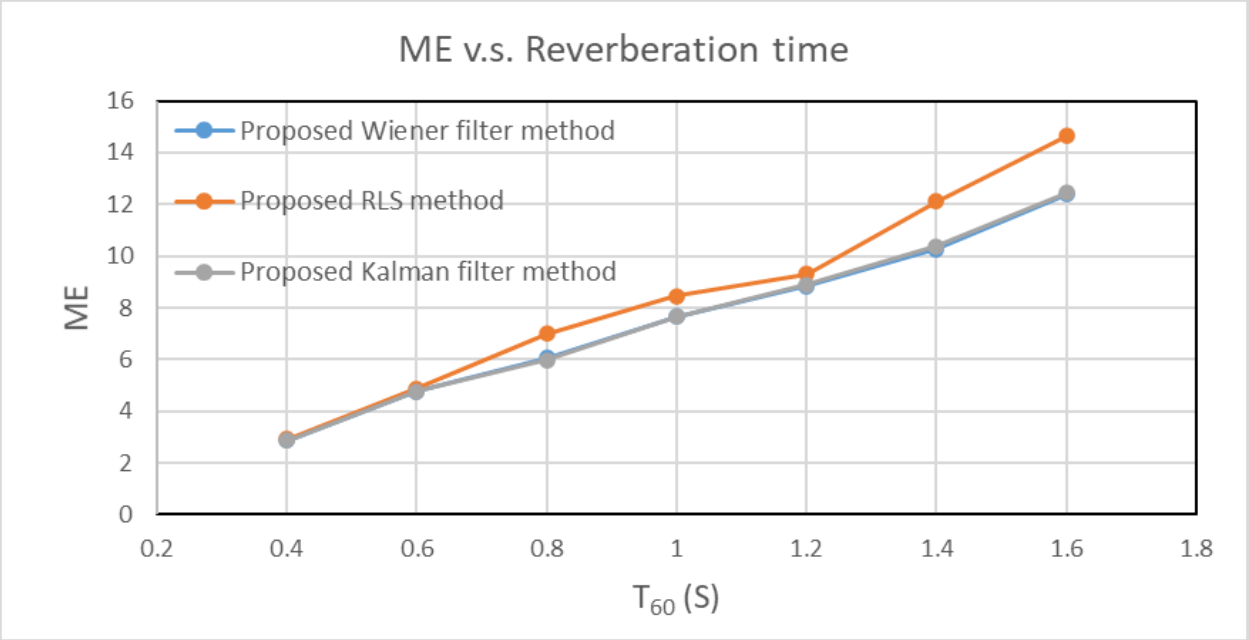


Figure 5 ME of the estimated ATFs based on the proposed methods in various reverberation times

Five types of signals were evaluated for their dereverberation performance metrics using the Perceptual Evaluation of Speech Quality (PESQ) [19] and Signal-to-Distortion Ratio (SDR) [20] . These included MINT dereverberation using RIRs estimated from the proposed method with a Wiener filter, MINT dereverberation using RIRs estimated from the proposed method with an RLS algorithm, MINT dereverberation using RIRs estimated from the proposed method with a Kalman stationary filter, dereverberation using WPE following DAS beamformer and unprocessed signals. Figure 6 depicts the acquired PESQ for seven varying reverberation times. The plot reveals that the three proposed methods outperform WPE following DAS beamformer in yielding a higher PESQ. It is noticeable that as the

reverberation time increases, the PESQs of five signals decline. In a similar vein, Figure 7 showcases the SDR obtained for the same seven reverberation times. It is clear from this graph that the proposed three methods continue to achieve higher SDR than WPE following DAS beamformer, which is a convincing outcome. It is also evident that the SDRs of all five signals decrease as the reverberation time increases.

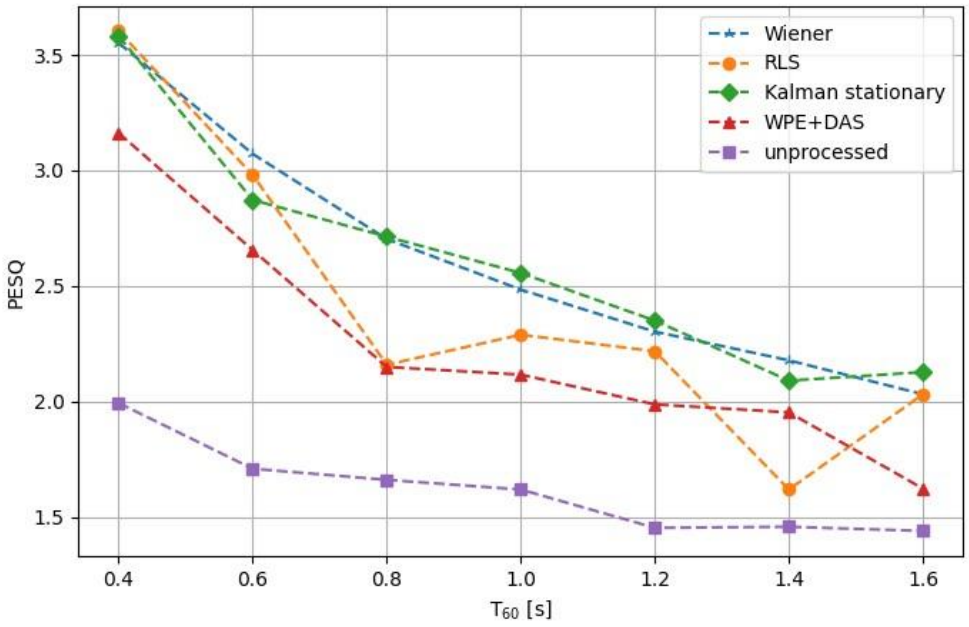


Figure 6 The PESQ values of the processed and unprocessed signals at different reverberation time

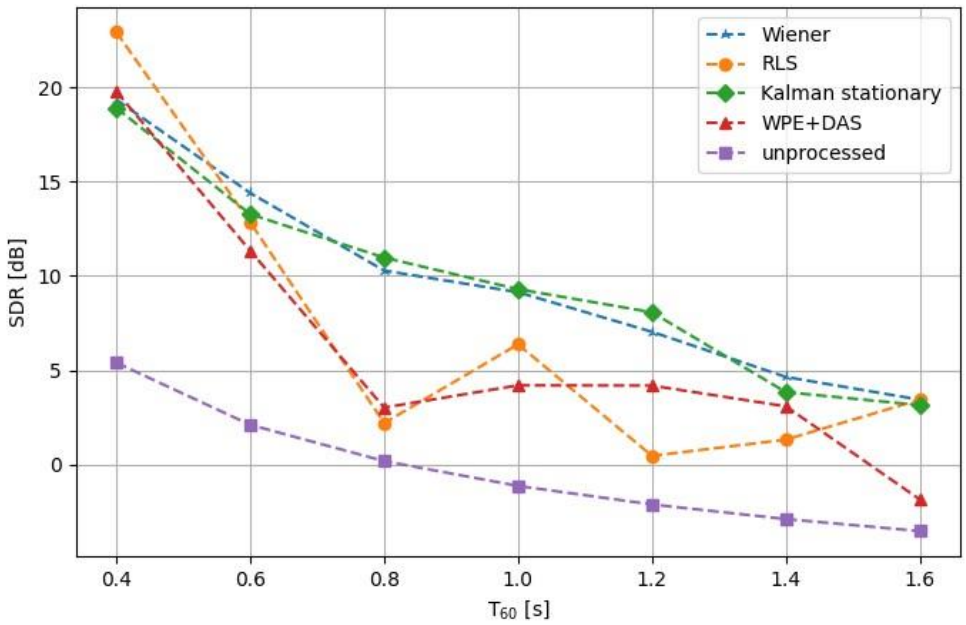


Figure 7 The SDR values of the processed and unprocessed signals at different reverberation time

4.2.2 With Parameters Optimization

Table 3 shows the ME of the estimated ATF of the 10-*th* microphone using the Kalman stationary filter with and without optimization when T_{60} is 0.6 seconds. The parameters of the Kalman stationary filter to be optimized are η and ρ. The parameters of the PSO, namely U, J, T_{max}, w_{in}, c_1 and c_2, are set to 2, 50, 100, 0.6, 2 and 2, respectively, and the parameters of the ASPSO, namely U, J, T_{max}, z_1, C_{in}, w_{max}, w_{min}, b, c_1 and c_2, are set to 2, 50, 100, 0.4, 4, 0.9, 0.4, 0.3, 2 and 2, respectively. Table 3 reveals that when the filter parameters are optimized using either PSO or ASPSO, the ME can reach a lower value, which is a preferable result.

Table 3 ME of the estimated ATFs with and without optimization at T_{60} = 0.6s

Kalman filter without parameters optimization	5.0698
Kalman filter with PSO	4.9672
Kalman filter with ASPSO	4.9432

Chapter 5. CONCLUSIONS AND FUTURE WORK

5.1. Conclusions

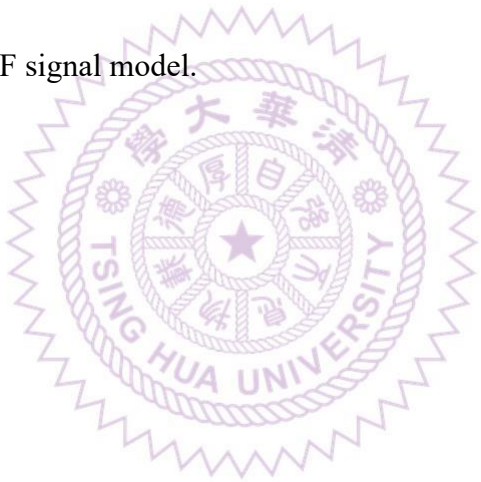
This paper presents a blind estimation method for the ATF based on the CTF model. Three techniques are developed for estimating CTF coefficient matrices using the Wiener filter, the RLS algorithm and Kalman stationary filter, respectively. By comparing the magnitude of the estimated ATF with the ground truth ATF and the ME over different reverberation times, it is concluded that the proposed method achieves accurate ATF estimation. Furthermore, using PESQ and SDR, we compare the results obtained from the dereverberation signal produced by WPE following DAS beamformer and MINT based on the estimated RIR of our proposed method. The results show a significant advantage of our proposed method over WPE following DAS beamformer. By optimizing the parameters used in the three proposed techniques, one can easily achieve lower ME of the estimated ATFs.

5.2. Future Work

Although in this paper we can accurately estimate the ATF for stationary sources, it is important to note that moving sources are more common in practical scenarios. Therefore, our next objective is to address the challenges posed by moving sources by exploiting the process noise characteristics introduced in the Kalman non-stationary

filter. Furthermore, all data presented in this article are the results of computer simulations. The inclusion of real experimental data for validation would increase the credibility of our results. Therefore, another future work is the design and implementation of real experiments.

Finally, although several state-of-the-art BSI techniques [5] [6] [7] have been shown to be useful in small reverberation time scenarios, they are still unable to tackle long reverberation time scenarios. Therefore, we will make efforts to modify these techniques using the CTF signal model.



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] 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.

[12] Harry L. Van Trees, “Optimum array processing: Part Iv of detection, estimation, and modulation theory,” *New York: Wiley*, 2002.

[13] 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.

[14] 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.

[15] R. E. Kalman, “A New Approach to Linear Filtering and Prediction Problems,” *Journal of Basic Engineering,* 1960.

[16] J. Kennedy, and R. Eberhart, “Particle swarm optimization,” *Proceedings of ICNN’95-International Conference on Neural Networks, IEEE,* pp. 1942-1948, 1995.

[17] 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.

[18] 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.

[19] 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.

[20] 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

[21] 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.

[22] 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.

[23] J. Wexler and S. Raz, "Discrete gabor expansions," *Signal Process.,* vol. 21, pp. 207–220, Nov. 1990.

[24] S. Qian and D. Chen, "Discrete Gabor transform," *IEEE Trans. Signal Process.,* vol. 41, no. 7, pp. 2429–2438, Jul. 1993.

[25] 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.

[26] 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.

[27] H. A. Hefny, and S. S. Azab, “Chaotic particle swarm optimization,” *The 7th International Conference on Informatics and Systems (INFOS),* pp. 1-8, 2010.

[28] Seyedali Mirjalili, and Andrew Lewis, “The whale optimization algorithm,” *Advances in Engineering Software,* vol. 95, pp.51-67, 2016.

[29] Emanuël Habets, “Room Impulse Response Generator,” *Internal Report,* pp. 1-17, 2006.