Kronecker Product Multichannel Linear Filtering for Adaptive Weighted Prediction Error-Based Speech Dereverberation

Gongping Huang, *Member, IEEE*, Jacob Benesty, Israel Cohen, *Fellow, IEEE*, and Jingdong Chen, *Fellow, IEEE*

***Abstract*—Reverberation, which is caused by late reflections, impairs not only speech quality but also intelligibility. Consequently, dereverberation, a process to mitigate the impact of reverberation, has attracted significant research interests. Numerous approaches have been developed in the literature, among which the weighted-prediction-error (WPE) one has demonstrated promising potential for reducing or eliminating reverberation. The WPE method has been well studied and several variants have been developed. The adaptive one, called adaptive WPE (AWPE) method, has been widely investigated for use in real applications as it can deal with reverberation in time-varying acoustic environments. However, the computational complexity of AWPE is high, which may be a problem for its implementation in real-time systems. This paper presents some new insights into AWPE-based speech dereverberation by introducing the concepts of Kronecker product and partially time-varying filtering. It then develops two algorithms for dereverberation with lower complexity than AWPE. The significant contributions of this work are as follows. First, we propose a Kronecker product filtering framework for speech dereverberation, where the linear prediction filter is formulated as the Kronecker product of two sets of shorter filters. Second, we propose a partially time-varying Kronecker product filter for dereverberation. Instead of estimating the entire linear prediction filter as in the conventional method, the proposed one only needs to update part of the filter. The proposed approaches can significantly reduce the computational complexity without sacrificing dereverberation performance as compared to AWPE. Simulation results validate the theoretical analysis and justify the advantages of the new methods.**

***Index Terms*—Adaptive weighted-prediction-error, kronecker product filtering, microphone arrays, multichannel linear prediction, speech dereverberation.**

Manuscript received September 10, 2021; revised February 2, 2022 and March 11, 2022; accepted March 16, 2022. Date of publication March 22, 2022; date of current version April 4, 2022. The work of Gongping Huang was supported by the Alexander von Humboldt Foundation. This work was supported in part by the Pazy Research Foundation, in part by the Alexander von Humboldt Foundation, in part by the National Key Research and Development Program of China under Grant 2018AAA0102200, and in part by the Key Program of National Science Foundation of China under Grants 61831019 and 62192713. The associate editor coordinating the review of this manuscript and approving it for publication was Prof. Stefania Cecchi. *(Corresponding author: Gongping Huang.)*

Gongping Huang was with the Technion - Israel Institute of Technology, Haifa 3200003, Israel. He is now with LMS, University of Erlangen-Nuremberg, 91058 Erlangen, Germany (e-mail: gongpinghuang@gmail.com).

Jacob Benesty is with the INRS-EMT, University of Quebec, Montreal, QC H5A 1K6, Canada (e-mail: Jacob.Benesty@inrs.ca).

Israel Cohen is with the Andrew and Erna Viterbi Faculty of Electrical and Computer Engineering, Technion–Israel Institute of Technology, Haifa 3200003, Israel (e-mail: icohen@ee.technion.ac.il).

Jingdong Chen is with the Center of Intelligent Acoustics and Immersive Communications and Shaanxi Provincial Key Laboratory of Artificial Intelligence, Northwestern Polytechnical University, Xi'an, Shaanxi 710072, China (e-mail: jingdongchen@ieee.org).

Digital Object Identifier 10.1109/TASLP.2022.3161150

I. Introduction

In acoustic environments with hands-free voice applications, the speech signal of interest picked up by microphones contains the direct-path component and attenuated and delayed replicas of the source speech signal. Reflections can be divided into early and late reflections depending on how much time it takes them to reach the microphones compared to the direct path. While early reflections are generally not harmful [1]–[3], late reflections form reverberation, which may severely impair both speech quality and intelligibility [4], [5]. Dereverberation, which is a process to exploit signal processing techniques to mitigate the impact of reverberation, has been widely studied [6]–[10]. Numerous approaches have been developed over the past few decades, such as channel equalization [11], [12], beamforming based methods [13]–[17], suppression based methods [18]–[20], and linear prediction based approaches [21], [22].

Among those approaches developed in the literature, the one based on multichannel linear prediction has demonstrated great potential for reducing or eliminating reverberation. This method first estimates late reflections with a delayed linear prediction filter and then subtracts the estimate from the observation. It has been widely studied, and various types of algorithms have been developed to implement this principle [23], [24], among which, the variance normalized delayed linear prediction algorithm, also known as the weighted-prediction-error (WPE) method, is shown to be very effective in reducing reverberation [25]–[31]. In real-time applications, the linear prediction filter needs to be estimated in an adaptive manner with, e.g., the recursive-least-squares (RLS) algorithm, [32]–[34] and the resulting method is called adaptive WPE (AWPE). However, the AWPE method is computationally costly, which makes it challenging to implement this technique in real systems. To reduce the complexity, a method was recently proposed to construct the linear prediction filter as a bilinear form (i.e., first-order Kronecker product) of a temporal filter and a spatial filter [35]. This method involves covariance matrices of much lower dimensions and is therefore computationally more efficient than the conventional WPE.

However, the spatial filter in this method requires such information as the room impulse responses or direction-of-arrivals (DOAs), which are generally unknown and have to be estimated.

To circumvent its drawbacks, this paper develops some new insights into the multichannel linear filtering-based approach to speech dereverberation by exploiting Kronecker product filtering. We express the linear prediction filter as a Kronecker product of two sets of shorter filters, which degenerates to the bilinear form when the order of the Kronecker product is one. We follow the conventional AWPE method to define a variance normalized cost function and derive an iterative RLS method to estimate the Kronecker product filters adaptively. Compared with the conventional AWPE that needs to estimate a long filter, the developed algorithm only needs to estimate two shorter filters. As a result, it has a much lower computational complexity as long as the order of the Kronecker decomposition is properly chosen.

Based on the proposed Kronecker product filtering framework, we propose an algorithm that expresses the filter as a Kronecker product of one set of time-invariant filters and one set of time-varying filters. Instead of updating a long linear prediction filter as the conventional AWPE method, we then derive a method that only updates the time-varying part of the filter so the computational efficiency is further improved while maintaining similar dereverberation performance.

The organization of this paper is as follows. Section II describes the signal model and problem formulation. Section III presents the proposed framework of Kronecker product multichannel linear prediction for dereverberation. Section IV introduces an algorithm based on partially time-varying Kronecker product filtering for dereverberation. Section VI analyzes the computational complexity of the different algorithms and compares their dereverberation performance. Conclusions are finally given in Section VII.

II. Signal Model and Problem Formulation

We consider the signal model in which a microphone array with M sensors captures a convolved source signal in some noise field. The received signal at the mth microphone is expressed as

$$y_m(k) = h_m(k) * s(k) + v_m(k),\ m = 1, 2, \ldots, M, \quad (1)$$

where $h_m(k)$ is the acoustic impulse response from the unknown speech source, $s(k)$, to the mth microphone, $*$ stands for the linear convolution, and $v_m(k)$ is the additive noise at the mth microphone. It is assumed that the convolved speech signals are coherent across the sensors, and the speech signals are uncorrelated with the noise signals. All the signals are assumed to be zero mean, real, and broadband.

In the short-time-Fourier-transform (STFT) domain, if we neglect the correlation across frequencies, the received signals can be well approximated as [31]

$$Y_m(n,\omega) = \sum_{l=0}^{J-1} H_m(l,\omega) S(n-l,\omega) + V_m(n,\omega), \quad m = 1, 2, \ldots M, \quad (2)$$

where n is the time-frame index, ω denotes the angular frequency, $H_m(n,\omega)$ of order J, is the counterpart of $h_m(k)$ in the STFT domain, $Y_m(n,\omega)$, $S(n,\omega)$, and $V_m(n,\omega)$ are the STFTs of $y_m(k)$, $s(k)$, and $v_m(k)$, respectively.

The multichannel linear prediction dereverberation process consists of estimating the late reflection component from the past L frames and then subtracting it from the observation to get an estimate of the source signal. Mathematically, this process is expressed as

$$\widehat{S}(n,\omega) = Y(n,\omega) - \mathbf{g}^H(\omega)\,\bar{\mathbf{y}}(n,\omega), \quad (3)$$

where $Y(n,\omega)$ is the reference signal (which can be chosen as the observation signal at any sensor), $\mathbf{g}(\omega)$ is the prediction filter of length $L_M = ML$, the subscript ^{H} is the conjugate-transpose operator, and

$$\bar{\mathbf{y}}(n,\omega) = \left[\mathbf{y}^T(n-D,\omega)\ \ \mathbf{y}^T(n-D-1,\omega) \cdots \mathbf{y}^T(n-D-L+1,\omega)\right]^T \quad (4)$$

is the stacked observation signal vector of length L_M, with

$$\mathbf{y}(n-D-l,\omega) = [Y_1(n-D-l,\omega)\ \ Y_2(n-D-l,\omega) \cdots Y_M(n-D-l,\omega)]^T,\ l = 0, 1, \ldots L-1 \quad (5)$$

being the observation signal vector of length M, the superscript ^{T} denotes the transpose of a vector or a matrix, and $D > 0$ is a predefined delay. Note that here we only consider a single output for simplicity and conciseness. The generalization to multiple outputs is straightforward.

Given the formulation in (3), the problem of dereverberation becomes one of finding the optimal filter, $\mathbf{g}(\omega)$, so that the late reflection components are suppressed as much as possible. One of the most widely used methods is WPE, where the filter is derived by maximizing the likelihood function of the speech and channel models [31]. In real-time applications, the dereverberation process is expressed as

$$\widehat{S}(n,\omega) = Y(n,\omega) - \mathbf{g}^H(n-1,\omega)\,\bar{\mathbf{y}}(n,\omega), \quad (6)$$

with $\mathbf{g}(n-1,\omega)$ being a time-varying filter that was updated at time frame $n-1$. The optimal linear prediction filter is generally derived based on the RLS adaptive method, which is often called the adaptive WPE (AWPE) method [32]. However, since it needs to update a long filter in every frequency band, the AWPE method is computationally expensive, making it challenging to implement in real-time applications.

III. Kronecker Product Linear Prediction for Dereverberation

The Kronecker product tool, which can decompose a long filter as a product of many short ones, has been successfully applied to many applications, such as beamforming [36]–[39], system identification [40], and echo cancellation [41]. In this study, we propose to apply this same technique to multichannel linear prediction for speech dereverberation. For this purpose, we write the linear prediction filter $\mathbf{g}(n,\omega)$ of length L_M as a

Kronecker product of P ($P \geq 1$) short filters [42]:

$$\mathbf{g}\left(n,\omega\right) = \sum_{p=1}^{P} \mathbf{g}_{2,p}\left(n,\omega\right) \otimes \mathbf{g}_{1,p}\left(n,\omega\right), \tag{7}$$

where $\otimes$ denotes the Kronecker product, $\mathbf{g}_{2,p}(n,\omega)$ and $\mathbf{g}_{1,p}(n,\omega)$, $p = 1, 2, \ldots, P$, are filters of lengths L_2 and L_1, respectively, with $L_M = L_2 L_1$. In our study, P is called the order of the Kronecker product filters. When $P = 1$, it degenerates to the bilinear form corresponding to the case in [35]. However, it should be pointed out that even with $P = 1$, we consider in this study a more general case rather than the one in [35] that limits the decomposition to a Kronecker product of a spatio-temporal filter. We can always set $P \leq \min(L_1, L_2)$ since it can be theoretical proved using the singular value decomposition (SVD) that any vector of length $L_M = L_2 L_1$ can be fully represented by $\min(L_1, L_2)$ or less than $\min(L_1, L_2)$ pairs of short filters of lengths L_2 and L_1, respectively [41], [42]. For ease of exposition and conciseness, we shall omit ω from the notation unless otherwise specified in the rest of this paper, which should not lead to any confusion.

For the Kronecker product, we have the following relationships [43]:

$$\begin{aligned}\mathbf{g}_{2,p}(n) \otimes \mathbf{g}_{1,p}(n) &= \left[\mathbf{g}_{2,p}(n) \otimes \mathbf{I}_{L_1}\right] \mathbf{g}_{1,p}(n) \\ &= \mathbf{G}_{2,p}(n)\mathbf{g}_{1,p}(n),\ p = 1, 2, \ldots, P, \end{aligned} \tag{8}$$

$$\begin{aligned}\mathbf{g}_{2,p}(n) \otimes \mathbf{g}_{1,p}(n) &= \left[\mathbf{I}_{L_2} \otimes \mathbf{g}_{1,p}(n)\right] \mathbf{g}_{2,p}(n) \\ &= \mathbf{G}_{1,p}(n)\mathbf{g}_{2,p}(n),\ p = 1, 2, \ldots, P, \end{aligned} \tag{9}$$

where $\mathbf{I}_{L_1}$ and $\mathbf{I}_{L_2}$ are the identity matrices of sizes $L_1 \times L_1$ and $L_2 \times L_2$, respectively, and

$$\mathbf{G}_{2,p}(n) = \mathbf{g}_{2,p}(n) \otimes \mathbf{I}_{L_1},\ p = 1, 2, \ldots, P, \tag{10}$$

$$\mathbf{G}_{1,p}(n) = \mathbf{I}_{L_2} \otimes \mathbf{g}_{1,p}(n),\ p = 1, 2, \ldots, P, \tag{11}$$

are matrices of sizes $L_1 L_2 \times L_1$ and $L_1 L_2 \times L_2$, respectively.

Clearly, at time frame n, the filters obtained at time frame $n-1$ are accessible, so the dereverberated signal can be written as

$$\widehat{S}(n) = Y(n) - \left[\sum_{p=1}^{P} \mathbf{g}_{2,p}\left(n-1\right) \otimes \mathbf{g}_{1,p}\left(n-1\right)\right]^H \bar{\mathbf{y}}(n). \tag{12}$$

Let us first assume that $\mathbf{g}_{2,p}(n-1),\ p = 1, 2, \ldots, P$, are fixed. Substituting (8) into (12), the dereverberated signal can be written as

$$\begin{aligned}\widehat{S}_1(n) &= Y(n) - \sum_{p=1}^{P} \mathbf{g}_{1,p}^H\left(n-1\right) \mathbf{G}_{2,p}^H\left(n-1\right) \bar{\mathbf{y}}(n) \\ &= Y(n) - \sum_{p=1}^{P} \mathbf{g}_{1,p}^H\left(n-1\right) \mathbf{y}_{2,p}(n) \\ &= Y(n) - \underline{\mathbf{g}}_1^H\left(n-1\right) \underline{\mathbf{y}}_2(n), \end{aligned} \tag{13}$$

where $\mathbf{G}_{2,p}(n-1) = \mathbf{g}_{2,p}(n-1) \otimes \mathbf{I}_{L_1}$ is a matrix of size $L_1 L_2 \times L_1$,

$$\mathbf{y}_{2,p}(n) = \mathbf{G}_{2,p}^H\left(n-1\right) \bar{\mathbf{y}}(n),\ p = 1, 2, \ldots, P, \tag{14}$$

is a vector of length L_1, and

$$\begin{aligned}\underline{\mathbf{g}}_1\left(n-1\right) = \big[&\mathbf{g}_{1,1}^T\left(n-1\right)\ \mathbf{g}_{1,2}^T\left(n-1\right) \\ &\cdots \mathbf{g}_{1,P}^T\left(n-1\right)\big]^T, \end{aligned} \tag{15}$$

$$\underline{\mathbf{y}}_2(n) = \left[\mathbf{y}_{2,1}^T(n)\ \mathbf{y}_{2,2}^T(n) \cdots\ \mathbf{y}_{2,P}^T(n)\right]^T \tag{16}$$

are vectors of length PL_1.

We now assume that $\mathbf{g}_{1,p}(n-1)$, $p = 1, 2, \ldots, P$, are fixed. Substituting (9) into (12), the dereverberated signal can then be written as

$$\begin{aligned}\widehat{S}_2(n) &= Y(n) - \sum_{p=1}^{P} \mathbf{g}_{2,p}^H\left(n-1\right) \mathbf{G}_{1,p}^H\left(n-1\right) \bar{\mathbf{y}}(n) \\ &= Y(n) - \sum_{p=1}^{P} \mathbf{g}_{2,p}^H\left(n-1\right) \mathbf{y}_{1,p}(n) \\ &= Y(n) - \underline{\mathbf{g}}_2^H\left(n-1\right) \underline{\mathbf{y}}_1(n), \end{aligned} \tag{17}$$

where $\mathbf{G}_{1,p}(n-1) = \mathbf{I}_{L_2} \otimes \mathbf{g}_{1,p}(n-1)$ is a matrix of size $L_1 L_2 \times L_2$,

$$\mathbf{y}_{1,p}(n) = \mathbf{G}_{1,p}^H(n)\bar{\mathbf{y}}(n),\ p = 1, 2, \ldots, P, \tag{18}$$

is a vector of length L_2, and

$$\begin{aligned}\underline{\mathbf{g}}_2\left(n-1\right) = \big[&\mathbf{g}_{2,1}^T\left(n-1\right)\ \mathbf{g}_{2,2}^T\left(n-1\right) \\ &\cdots\ \mathbf{g}_{2,P}^T\left(n-1\right)\big]^T, \end{aligned} \tag{19}$$

$$\underline{\mathbf{y}}_1(n) = \left[\mathbf{y}_{1,1}^T(n)\ \mathbf{y}_{1,2}^T(n) \cdots\ \mathbf{y}_{1,P}^T(n)\right]^T \tag{20}$$

are vectors of length PL_2.

As seen, the problem of estimating the multichannel linear prediction filter, $\mathbf{g}(n)$, of length L_M, is now reformulated as one of estimating two shorter filters, $\underline{\mathbf{g}}_1(n)$ and $\underline{\mathbf{g}}_2(n)$, of lengths PL_1 and PL_2, respectively. Since the two filters are coupled with each other, it is challenging to estimate both in one step. In what follows, we will present a two-step method to estimate them.

Following the widely used AWPE method [31] and the RLS algorithm developed in [44], we define the variance normalized cost functions under the least-squares (LS) error criterion as [44]

$$\mathcal{J}\left[\underline{\mathbf{g}}_1(n)|\underline{\mathbf{g}}_2(n)\right] = \sum_{i=1}^{n} \alpha_1^{n-i} \frac{\left|Y(i) - \underline{\mathbf{g}}_1^H(n)\underline{\mathbf{y}}_2(i)\right|^2}{\lambda_1(i)}, \tag{21}$$

$$\mathcal{J}\left[\underline{\mathbf{g}}_2(n)|\underline{\mathbf{g}}_1(n)\right] = \sum_{i=1}^{n} \alpha_2^{n-i} \frac{\left|Y(i) - \underline{\mathbf{g}}_2^H(n)\underline{\mathbf{y}}_1(i)\right|^2}{\lambda_2(i)}, \tag{22}$$

where α_1 ($0 < \alpha_1 < 1$) and α_2 ($0 < \alpha_2 < 1$) are forgetting factors, and

$$\lambda_1(i) = \left|\widehat{S}_1(n)\right|^2,\ i = 1, 2, \ldots, n, \tag{23}$$

$$\lambda_2(i) = \left|\widehat{S}_2(n)\right|^2,\ i = 1, 2, \ldots, n, \tag{24}$$

are estimates of the short-time variance of the desired speech signal.[1] Note that in the optimization criterion for $\underline{\mathbf{g}}_1(n)$, we assume that all $\underline{\mathbf{g}}_1(i)$, $i = 1, 2, \ldots, n-1$, are fixed, and for $\underline{\mathbf{g}}_2(n)$, all $\underline{\mathbf{g}}_2(i)$, $i = 1, 2, \ldots, n-1$, are fixed [44].

The cost function in (21) can be expanded as

$$\begin{aligned}\mathcal{J}\left[\underline{\mathbf{g}}_1(n)|\underline{\mathbf{g}}_2(n)\right] &= \phi(n) - \underline{\mathbf{g}}_1^H(n)\boldsymbol{\rho}_{\underline{\mathbf{y}}_2}(n) \\ &\quad - \boldsymbol{\rho}_{\underline{\mathbf{y}}_2}^H(n)\underline{\mathbf{g}}_1(n) \\ &\quad + \underline{\mathbf{g}}_1^H(n)\boldsymbol{\Phi}_{\underline{\mathbf{y}}_2}(n)\underline{\mathbf{g}}_1(n),\end{aligned} \tag{25}$$

where

$$\phi(n) = \sum_{i=1}^{n} \alpha_1^{n-i} \frac{|Y(i)|^2}{\lambda_1(i)} \tag{26}$$

is the weighted variance of the reference signal, and

$$\boldsymbol{\rho}_{\underline{\mathbf{y}}_2}(n) = \sum_{i=1}^{n} \alpha_1^{n-i} \frac{\underline{\mathbf{y}}_2(i)Y^*(i)}{\lambda_1(i)}, \tag{27}$$

$$\boldsymbol{\Phi}_{\underline{\mathbf{y}}_2}(n) = \sum_{i=1}^{n} \alpha_1^{n-i} \frac{\underline{\mathbf{y}}_2(i)\underline{\mathbf{y}}_2^H(i)}{\lambda_1(i)} \tag{28}$$

are the weighted correlation vector of length PL_1 and weighted cross-correlation matrix of size $PL_1 \times PL_1$.

Minimization of $\mathcal{J}[\underline{\mathbf{g}}_1(n)|\underline{\mathbf{g}}_2(n)]$ with respect to $\underline{\mathbf{g}}_1(n)$ leads to the optimal solution:

$$\underline{\mathbf{g}}_1(n) = \boldsymbol{\Phi}_{\underline{\mathbf{y}}_2}^{-1}(n)\boldsymbol{\rho}_{\underline{\mathbf{y}}_2}(n), \tag{29}$$

where the weighted correlation vector $\boldsymbol{\rho}_{\underline{\mathbf{y}}_2}(n)$ and cross-correlation matrix $\boldsymbol{\Phi}_{\underline{\mathbf{y}}_2}(n)$ can be written in a recursive way as

$$\boldsymbol{\rho}_{\underline{\mathbf{y}}_2}(n) = \alpha_1\boldsymbol{\rho}_{\underline{\mathbf{y}}_2}(n-1) + \frac{\underline{\mathbf{y}}_2(n)Y^*(n)}{\lambda_1(n)}, \tag{30}$$

$$\boldsymbol{\Phi}_{\underline{\mathbf{y}}_2}(n) = \alpha_1\boldsymbol{\Phi}_{\underline{\mathbf{y}}_2}(n-1) + \frac{\underline{\mathbf{y}}_2(n)\underline{\mathbf{y}}_2^H(n)}{\lambda_1(n)}. \tag{31}$$

Using the Woodbury's identity (also known as matrix inversion lemma), the inverse weighted cross-correlation matrix $\boldsymbol{\Phi}_{\underline{\mathbf{y}}_2}^{-1}(n)$ can be recursively estimated as

$$\begin{aligned}\boldsymbol{\Phi}_{\underline{\mathbf{y}}_2}^{-1}(n) = \frac{1}{\alpha_1}\Big[&\boldsymbol{\Phi}_{\underline{\mathbf{y}}_2}^{-1}(n-1) \\ &-\boldsymbol{\kappa}_2(n)\underline{\mathbf{y}}_2^H(n)\boldsymbol{\Phi}_{\underline{\mathbf{y}}_2}^{-1}(n-1)\Big],\end{aligned} \tag{32}$$

where $\mathbf{I}_{PL_1}$ denotes the identity matrix of size $PL_1 \times PL_1$, and

$$\boldsymbol{\kappa}_2(n) = \frac{\boldsymbol{\Phi}_{\underline{\mathbf{y}}_2}^{-1}(n-1)\,\underline{\mathbf{y}}_2(n)}{\alpha_1\lambda_1(n) + \underline{\mathbf{y}}_2^H(n)\boldsymbol{\Phi}_{\underline{\mathbf{y}}_2}^{-1}(n-1)\,\underline{\mathbf{y}}_2(n)} \tag{33}$$

[1]The WPE method is derived from the maximization of the log-likelihood function of the speech and channel models, where the linear prediction filter coefficients and the variance of the desired speech signal are optimized alternately [31]. When the linear prediction filter is obtained, the optimal estimation of the variance of the desired speech signal is estimated according to (23) and (24).

Algorithm 1: The KAWPE Algorithm.

Initialize $\underline{\mathbf{g}}_1(0)$, $\underline{\mathbf{g}}_2(0)$, $\boldsymbol{\Phi}_{\underline{\mathbf{y}}_1}^{-1}(0)$, $\boldsymbol{\Phi}_{\underline{\mathbf{y}}_2}^{-1}(0)$
for $n = 1, 2, \ldots$ **do**
  Construct $\underline{\mathbf{y}}_2(n)$
  Calculate the dereverberated signal and its variance
  $\widehat{S}_1(n) = Y(n) - \underline{\mathbf{g}}_1^H(n-1)\underline{\mathbf{y}}_2(n)$
  $\lambda_1(n) = |\widehat{S}_1(n)|^2$
  Calculate the gain vector
  $\boldsymbol{\kappa}_2(n) = \frac{\boldsymbol{\Phi}_{\underline{\mathbf{y}}_2}^{-1}(n-1)\underline{\mathbf{y}}_2(n)}{\alpha_1\lambda_1(n)+\underline{\mathbf{y}}_2^H(n)\boldsymbol{\Phi}_{\underline{\mathbf{y}}_2}^{-1}(n-1)\underline{\mathbf{y}}_2(n)}$
  update the inverse weighted cross-correlation matrix
  $\boldsymbol{\Phi}_{\underline{\mathbf{y}}_2}^{-1}(n) = \frac{1}{\alpha_1}[\boldsymbol{\Phi}_{\underline{\mathbf{y}}_2}^{-1}(n-1) - \boldsymbol{\kappa}_2(n)\underline{\mathbf{y}}_2^H(n)\boldsymbol{\Phi}_{\underline{\mathbf{y}}_2}^{-1}(n-1)]$
  update the prediction filter
  $\underline{\mathbf{g}}_1(n) = \underline{\mathbf{g}}_1(n-1) + \boldsymbol{\kappa}_2(n)\widehat{S}_1^*(n)$
  Construct $\underline{\mathbf{y}}_1(n)$
  Calculate the dereverberated signal and its variance
  $\widehat{S}_2(n) = Y(n) - \underline{\mathbf{g}}_2^H(n-1)\underline{\mathbf{y}}_1(n)$
  $\lambda_2(n) = |\widehat{S}_2(n)|^2$
  Calculate the gain vector
  $\boldsymbol{\kappa}_1(n) = \frac{\boldsymbol{\Phi}_{\underline{\mathbf{y}}_1}^{-1}(n-1)\underline{\mathbf{y}}_1(n)}{\alpha_2\lambda_2(n)+\underline{\mathbf{y}}_1^H(n)\boldsymbol{\Phi}_{\underline{\mathbf{y}}_1}^{-1}(n-1)\underline{\mathbf{y}}_1(n)}$
  update the inverse weighted cross-correlation matrix
  $\boldsymbol{\Phi}_{\underline{\mathbf{y}}_1}^{-1}(n) = \frac{1}{\alpha_2}[\boldsymbol{\Phi}_{\underline{\mathbf{y}}_1}^{-1}(n-1) - \boldsymbol{\kappa}_1(n)\underline{\mathbf{y}}_1^H(n)\boldsymbol{\Phi}_{\underline{\mathbf{y}}_1}^{-1}(n-1)]$
  update the prediction filter
  $\underline{\mathbf{g}}_2(n) = \underline{\mathbf{g}}_2(n-1) + \boldsymbol{\kappa}_1(n)\widehat{S}_2^*(n)$
end for

is a gain vector of length PL_1.

Substituting (30) and (32) into (29) gives a recursive form of updating the filter $\underline{\mathbf{g}}_1(n)$, i.e.,

$$\underline{\mathbf{g}}_1(n) = \underline{\mathbf{g}}_1(n-1) + \boldsymbol{\kappa}_2(n)\widehat{S}_1^*(n). \tag{34}$$

In a similar way, the filter $\underline{\mathbf{g}}_2(n)$ can be recursively updated as

$$\underline{\mathbf{g}}_2(n) = \underline{\mathbf{g}}_2(n-1) + \boldsymbol{\kappa}_1(n)\widehat{S}_2^*(n), \tag{35}$$

where

$$\boldsymbol{\kappa}_1(n) = \frac{\boldsymbol{\Phi}_{\underline{\mathbf{y}}_1}^{-1}(n-1)\,\underline{\mathbf{y}}_1(n)}{\alpha_2\lambda_2(n) + \underline{\mathbf{y}}_1^H(n)\boldsymbol{\Phi}_{\underline{\mathbf{y}}_1}^{-1}(n-1)\,\underline{\mathbf{y}}_1(n)} \tag{36}$$

is a vector of length PL_2, and the inverse weighted cross-correlation matrix $\boldsymbol{\Phi}_{\underline{\mathbf{y}}_1}^{-1}(n)$ is recursively estimated as

$$\begin{aligned}\boldsymbol{\Phi}_{\underline{\mathbf{y}}_1}^{-1}(n) = \frac{1}{\alpha_2}\Big[&\boldsymbol{\Phi}_{\underline{\mathbf{y}}_1}^{-1}(n-1) \\ &-\boldsymbol{\kappa}_1(n)\underline{\mathbf{y}}_1^H(n)\boldsymbol{\Phi}_{\underline{\mathbf{y}}_1}^{-1}(n-1)\Big].\end{aligned} \tag{37}$$

Consequently, the two filters $\underline{\mathbf{g}}_1(n)$ and $\underline{\mathbf{g}}_2(n)$ are iteratively updated and $\widehat{S}_2(n)$ is considered as the final output. To be consistent with the conventional AWPE algorithm [31], [32], we call the proposed method as the Kronecker product AWPE (KAWPE).

The KAWPE algorithm is summarized in Algorithm 1. As noticed, the conventional AWPE method aims at estimating the filter, $\mathbf{g}(n)$, of length L, while the proposed KAWPE method deals with the estimation of two shorter filters, $\underline{\mathbf{g}}_1(n)$ and $\underline{\mathbf{g}}_2(n)$, of lengths PL_1 and PL_2, respectively. Consequently, the computational complexity of the KAWPE algorithm can be significantly lower than that of AWPE, which will be further discussed in Section V.

IV. Partially Time-Varying Kronecker Product Linear Prediction for Dereverberation

Based on the proposed Kronecker product filters, we propose in what follows a method that updates the time-varying Kronecker product filter for speech dereverberation. In contrast with the conventional AWPE algorithm that estimates the entire linear prediction filter, we propose to update only part of the prediction filters. Let us write the linear prediction filter of length L_M as

$$\begin{aligned}\mathbf{g}(n) &= \sum_{p=1}^{P} \underbrace{\mathbf{g}_{2,p}}_{\text{time-invariant}} \otimes \underbrace{\mathbf{g}_{1,p}(n)}_{\text{time-varying}} \\ &= \sum_{p=1}^{P} \left(\mathbf{g}_{2,p} \otimes \mathbf{I}_{L_1}\right) \mathbf{g}_{1,p}(n) \\ &= \sum_{p=1}^{P} \mathbf{G}_{2,p}\mathbf{g}_{1,p}(n), \end{aligned} \tag{38}$$

where $\mathbf{g}_{2,p},\ p = 1, 2, \ldots, P$, are time-invariant filters of length L_2, $\mathbf{g}_{1,p}(n),\ p = 1, 2, \ldots, P$, are time-varying filters of length L_1, and $\mathbf{G}_{2,p} = \mathbf{g}_{2,p} \otimes \mathbf{I}_{L_1},\ p = 1, 2, \ldots, P$, are also time-invariant.

The time-invariant filters can be estimated in advance, and only time-varying filters need to be estimated adaptively. This study illustrates the basic concept of using a partially time-varying Kronecker product filter for speech dereverberation. So, we consider a simple strategy that uses the first few seconds of the signal to compute the KAWPE filters according to Algorithm 1, and then keeps the estimated $\underline{\mathbf{g}}_2(n)$, or the corresponding $\mathbf{g}_{2,p},\ p = 1, 2, \ldots, P$, as the time-invariant filters. The time-invariant filters can be optimized or learned with a better strategy, which is worth further study.

Now, the dereverberated signal can be written as

$$\begin{aligned}\widehat{S}(n) &= Y(n) - \left[\sum_{p=1}^{P} \mathbf{g}_{2,p} \otimes \mathbf{g}_{1,p}\left(n-1\right)\right]^H \bar{\mathbf{y}}(n) \\ &= Y(n) - \sum_{p=1}^{P} \mathbf{g}_{1,p}^H\left(n-1\right) \mathbf{G}_{2,p}^H \bar{\mathbf{y}}(n) \\ &= Y(n) - \sum_{p=1}^{P} \mathbf{g}_{1,p}^H\left(n-1\right) \mathbf{y}_{2,p}(n) \\ &= Y(n) - \underline{\mathbf{g}}_1^H\left(n-1\right) \underline{\mathbf{y}}_2(n), \end{aligned} \tag{39}$$

Algorithm 2: The PTV-AWPE Algorithm.

```
Set time-invariant filters g_{2,p} for p = 1, 2, ..., P
Initialize g_1(0) and Φ^{-1}_{y_2}(0)
for n = 1, 2, ... do
    Construct y_2(n)
    Calculate the dereverberated signal and its variance
    Ŝ_1(n) = Y(n) − g_1^H(n−1) y_2(n)
    λ_1(n) = |Ŝ_1(n)|^2
    Calculate vector
    κ_2(n) = Φ^{-1}_{y_2}(n−1) y_2(n) / (α_1 λ_1(n) + y_2^H(n) Φ^{-1}_{y_2}(n−1) y_2(n))
    update the inverse weighted cross-correlation matrix
    Φ^{-1}_{y_2}(n) =
    (1/α_1)[Φ^{-1}_{y_2}(n−1) − κ_2(n) y_2^H(n) Φ^{-1}_{y_2}(n−1)]
    update the prediction filter
    g_1(n) = g_1(n−1) + κ_2(n) Ŝ_1^*(n)
end for
```

where

$$\mathbf{y}_{2,p}(n) = \mathbf{G}_{2,p}^H \bar{\mathbf{y}}(n),\ p = 1, 2, \ldots, P, \tag{40}$$

and $\underline{\mathbf{g}}_1(n-1)$ and $\underline{\mathbf{y}}_2(n)$ are defined analogously to (15) and (16).

Then, we can define the cost function under the LS error criterion and estimate the filter recursively in a similar way as in Section III. The derivation process is almost the same as that in Section III and the only major difference is that no iterative optimization is needed here since we only need to update the time-varying filters $\underline{\mathbf{g}}_1(n)$. We omit the detailed derivation to keep the paper concise and only summarize the proposed method in Algorithm 2. We call the proposed method a partially time-varying Kronecker product AWPE (PTV-KAWPE) method.

V. Computational Complexity Analysis

As discussed in Section III, the conventional AWPE method attempts to estimate the filter, $\mathbf{g}(n)$ of length L, while the proposed KAWPE method estimates the two shorter filters, $\underline{\mathbf{g}}_1(n)$ and $\underline{\mathbf{g}}_2(n)$ of lengths PL_1 and PL_2, respectively. Estimating shorter filters can help greatly reduce the computational complexity, which is investigated in this section. We will compare the complexity of the proposed KAWPE and PTV-KAWPE methods with that of the conventional AWPE method [31].

One important step in KAWPE is the construction of the signal vectors $\underline{\mathbf{y}}_2(n)$ and $\underline{\mathbf{y}}_1(n)$, which are not computationally efficient to construct directly according to (14) and (18). Through some analysis, one can check that the vector $\mathbf{y}_{2,p}(n)$ defined in (14) can be computed as

$$\mathbf{y}_{2,p}(n) = \mathbf{Y}(n)\mathbf{g}_{2,p}^*\left(n-1\right),\ p = 1, 2, \ldots, P, \tag{41}$$

where

$$\mathbf{Y}(n) = \begin{bmatrix} Y_1(n) & Y_{L_1+1}(n) & \cdots & Y_{(L_2-1)L_1+1}(n) \\ Y_2(n) & Y_{L_1+2}(n) & \cdots & Y_{(L_2-1)L_1+2}(n) \\ \vdots & \vdots & \ddots & \vdots \\ Y_{L_1}(n) & Y_{2L_1}(n) & \cdots & Y_{L_2L_1}(n) \end{bmatrix}$$

TABLE I
COMPUTATIONAL COMPLEXITY IN TERMS OF COMPLEX-VALUED MULTIPLICATIONS OF THE CONVENTIONAL AWPE METHOD, THE PROPOSED KAWPE METHOD, AND THE PROPOSED PTV-KAWPE METHOD

,

Method	Complex-valued Multiplications	Computational Complexity
AWPE	$4L_M^2+4L_M+3=4L_1^2L_2^2+4L_1L_2+3$	$\sim O(L_1^2L_2^2)$
KAWPE	$4P^2(L_1^2+L_2^2)+2PL_1L_2+4P(L_1+L_2)+6$	$\sim O(P^2L_1^2+P^2L_2^2)$
PTV-KAWPE	$4P^2L_1^2+PL_1L_2+4PL_1+3$	$\sim O(P^2L_1^2+P^2L_2^2)$

is a matrix of size $L_1 \times L_2$, with $Y_i(n)$ being the ith ($i = 1, 2, \ldots, L_2L_1$) element of the vector $\bar{\mathbf{y}}(n)$ defined in (4). It is seen clearly from (41) that the computation of $\mathbf{y}_{2,p}(n)$ needs PL_2L_1 complex-valued multiplications every time. Similarly, the construction of the signal $\underline{\mathbf{y}}_2(n)$ needs PL_2L_1 complex-valued multiplications.

Following the same analysis, one can arrange the vector $\mathbf{y}_{1,p}(n)$ defined in (18) as

$$\mathbf{y}_{1,p}(n) = \mathbf{Y}^T(n)\mathbf{g}_{1,p}^*(n-1)\,,\ p = 1, 2, \ldots, P. \tag{42}$$

It follows immediately that the construction of the signal $\underline{\mathbf{y}}_1(n)$ also needs PL_2L_1 multiplications.

Now, let us analyze the computational complexity related to the estimation of the filter $\underline{\mathbf{g}}_1(n)$. First, as shown in (15), the estimation of $\widehat{S}(n)$ needs PL_1 multiplications. Then, the calculation of the vector $\boldsymbol{\kappa}_2(n)$ needs $P^2L_1^2+2PL_1+3$ multiplications, where the division in (33) needs equivalently 1 division and PL_1 multiplications (let us we approximate it as PL_1+1 multiplications). Next, the update of the inverse weighted cross-correlation matrix $\boldsymbol{\Phi}_{\underline{\mathbf{y}}_2}^{-1}(n)$ needs $3P^2L_1^2$ multiplications ($1/\alpha_2$ can be computed in advance). Finally, PL_1 multiplications are required to update the prediction filter $\underline{\mathbf{g}}_1(n)$. So, a total of $4P^2L_1^2+4PL_1+3$ multiplications are needed in order to estimate $\underline{\mathbf{g}}_1(n)$. Similarly, one can check that the total number of multiplications needed to estimate $\underline{\mathbf{g}}_2(n)$, which is $4P^2L_2^2+4PL_2+3$. Consequently, a total of $4P^2(L_1^2+L_2^2)+2PL_1L_2+4P(L_1+L_2)+6$ complex-valued multiplications are needed for the proposed KAWPE method. Note that in the aforementioned analysis of complexity, we neglected the number of additions/subtractions as they require less computational time than multiplications/divisions. Also, we did not use $\mathbf{x}^H\boldsymbol{\Phi} = (\boldsymbol{\Phi}\mathbf{x})^H$ as in the conventional AWPE method [32] to avoid numerical problems [the conventional AWPE method uses $\mathbf{x}^H\boldsymbol{\Phi} = (\boldsymbol{\Phi}\mathbf{x})^H$ where $\boldsymbol{\Phi}^H = \boldsymbol{\Phi}$, $\mathbf{x} \in \{\underline{\mathbf{y}}_2(n), \underline{\mathbf{y}}_1(n), \bar{\mathbf{y}}(n)\}$, $\boldsymbol{\Phi} \in \{\boldsymbol{\Phi}_{\underline{\mathbf{y}}_2}^{-1}(n), \boldsymbol{\Phi}_{\underline{\mathbf{y}}_1}^{-1}(n), \boldsymbol{\Phi}_{\bar{\mathbf{y}}}^{-1}(n)\}$, and $\boldsymbol{\Phi}_{\bar{\mathbf{y}}}^{-1}(n)$ is the inverse weighted cross-correlation matrix].

In summary, the KAWPE method has a computational complexity proportional to $O(P^2L_1^2+P^2L_2^2)$. Following the same analysis, one can check that the PTV-KAWPE method needs $4P^2L_1^2+PL_1L_2+4PL_1+3$ complex-valued multiplications. In comparison, the conventional AWPE method needs $4L_M^2+4L_M+3=4L_1^2L_2^2+4L_1L_2+3$ complex-valued multiplications. Table I summarizes the computational complexity of AWPE, KAWPE, and PTV-KAWPE. It is seen that the proposed KAWPE and PTV-KAWPE algorithms have lower computational complexity as long as the parameters are appropriately chosen.

VI. SIMULATIONS AND EXPERIMENTS

In this section, we study the performance of the proposed methods for speech dereverberation. We first assess the performance of KAWPE and PTV-KAWPE in simulated room environments with different values of P. Then, we study and compare the performances of AWPE, KAWPE and PTV-KAWPE in real room environments where both reverberation and background noise coexist and the source of interest changes in position during the evaluation process. Finally, we compare the computational complexities of AWPE, KAWPE, and PTV-KAWPE.

A. Performance Study With Simulated Room Environments

The clean source speech signal used in this experiment was recorded in a quiet office room. The acoustic channel impulse responses from the source to the microphones are generated using the image model method [45], where the wall reflection coefficients control the level of reverberation. We consider a uniform linear array of 4 omnidirectional microphones located in a room of size $6\ \mathrm{m} \times 8\ \mathrm{m} \times 4\ \mathrm{m}$. The positions of the four microphones are, respectively, (x, 2.0, 1.6), where $x = 2.936 : 0.0213 : 3.0638$, and a desired speaker is located at (5.0, 4.0, 1.5). We set all the reflection coefficients of the six walls to 0.8, where the corresponding reverberation time, T_{60}, is approximately 400 ms. The microphone observation signals are generated by convolving the source signal with the corresponding acoustic impulse responses. All signals are sampled at 16 kHz. The dereverberation process is implemented in the STFT domain, where the observation signals are divided into overlapping frames (of size 512 sample) with 75% overlapping, and a Kaiser window is applied. After dereverberation in the STFT domain, the enhanced time-domain signal is reconstructed with an overlap-add method. To evaluate the performance of the dereverberation methods, we adopt four widely used measures for speech dereverberation [3], [46]: the cepstral distance (CD), the log-likelihood ratio (LLR), the frequency-weighted segmental SNR (FWSNR), and the perceptual evaluation of speech quality (PESQ) [47]. Generally, for CD and LLR, the smaller are the values, the better is the dereverberation performance. In contrast, for FWSNR and PESQ, the larger are the values, the better is the speech dereverberation performance. To explicitly show the performance improvement in comparison to the original reverberant signal, we define the gain in CD, LLR, FWSNR,

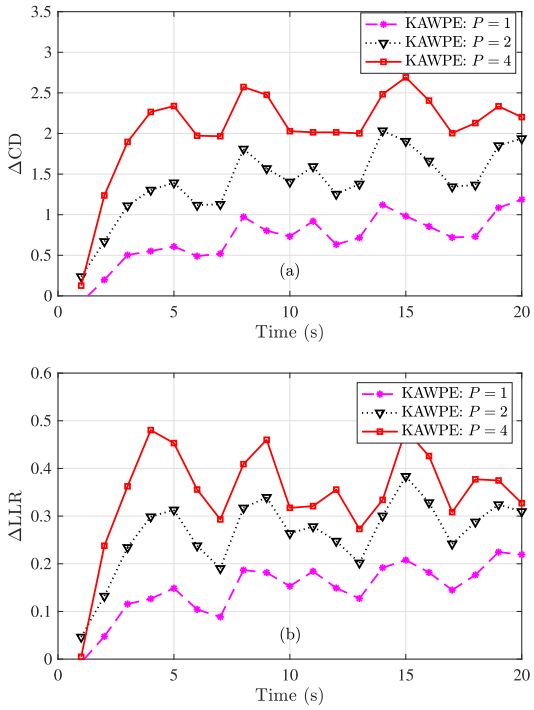


Fig. 1. Improvement of CD and LLR of the KAWPE method with different values of P vary with time: (a) ΔCD and (b) ΔLLR. Experimental conditions: $M=4$, $L_1=8$, and $L_2=8$.

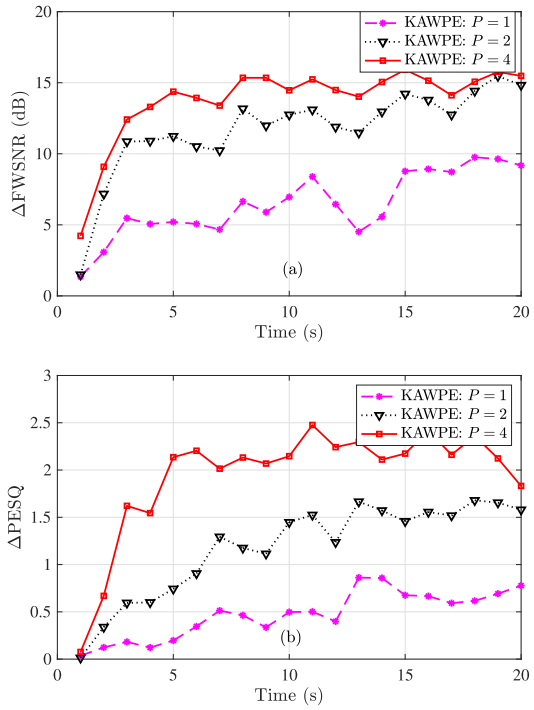


Fig. 2. Improvement of FWSNR and PESQ of the KAWPE method with different values of P vary with time: (a) ΔFWSNR and (b) ΔPESQ. Experimental conditions: $M=4$, $L_1=8$, and $L_2=8$.

and PESQ as

$$\Delta\text{CD} = \text{CD}_{\text{original}} - \text{CD},$$
$$\Delta\text{LLR} = \text{LLR}_{\text{original}} - \text{LLR},$$
$$\Delta\text{FWSNR} = \text{FWSNR} - \text{FWSNR}_{\text{original}}\ (\text{dB}),$$
$$\Delta\text{PESQ} = \text{PESQ} - \text{PESQ}_{\text{original}},$$

where the subscript "original" denotes the corresponding value of the measure of the original reverberant signal.

The parameters of KAWPE are set as follows: $L=16$, $M=4$, $L_M=64$, and $D=5$ (which indicates that the reflection paths in the acoustic impulse response before the first 40 ms are considered as early reflections) and the length of the two sets of Kronecker filters $\mathbf{g}_{2,p}(n)$ and $\mathbf{g}_{1,p}(n)$ are, respectively, $L_2=8$ and $L_1=8$. The recursive factors in KAWPE are set as $\alpha_1=\alpha_2=0.99$. The filters $\mathbf{g}_{2,p}(n)$ are initialized as $\mathbf{g}_{2,p}(n)=[0\ \cdots\ \epsilon\ \cdots\ 0]^T, p=1,2,\ldots,P$, where $\epsilon>0$ is the pth element of $\mathbf{g}_{2,p}$. In our implementation, we set $\epsilon=0.5$ unless otherwise specified. The filters $\mathbf{g}_{1,p}(n)$ are initialized as all-zero vectors. The inverse weighted cross-correlation matrices $\mathbf{\Phi}_{\underline{\mathbf{y}}_1}^{-1}(n)$ and $\mathbf{\Phi}_{\underline{\mathbf{y}}_2}^{-1}(n)$ are initialized as identity matrices.

We first study the performance of KAWPE with different values of P. To see how the performance varies with time, we divide the signal into short segments (each segment is 2-second long) and evaluate the performance for each segment. Fig. 1 shows plots of the CD and LLR gains of KAWPE with different values of $P\in\{1,2,4\}$ as a function of time. Fig. 2 shows plots of the FWSNR and PESQ gains of KAWPE with different values of $P\in\{1,2,4\}$ again as a function of time. As seen from Figs. 1 and 2, the application of KAWPE has significantly improved the CD, LLR, FWSNR and PESQ, indicating that this method is effective in performing dereverberation. It is also seen that the performance gain increases with the value of P, so a large value of P is preferred from the dereverberation performance perspective.

To see more clearly the impact of the order of the Kronecker filters, i.e., the value of P, on the dereverberation performance, we show in Figs. 3 and 4 the improvement of CD, LLR, FWSNR, and PESQ of KAWPE as a function of the order P with $M=2$, $L_1=6$, $L_2=6$ and $M=4$, $L_1=8$, $L_2=8$ where the rest of the conditions are same as in the previous experiment. As pointed out in Section III, the value of P should always satisfy $P\leq\min(L_1,L_2)$. As seen from Figs. 3 and 4, when $M=2$, the performance increases with P. When $M=4$, the performance first improves with P and then decreases, and the best performance appears at $P=4$ or 5, which corresponds to a significant improvement of speech quality. Note that the computational complexity of the algorithm also increases with the value of P.

In principle, initialization of the Kronecker product filters should not affect much the convergence and final performance of the developed algorithms as long as all the filters are not initialized to zero vectors, which would lead to trivial solutions. To validate this, we carried out one set of experiments with $L_1=8$, $L_2=8$, $P=3$, and the initialization is done as follows. The filters $\mathbf{g}_{1,p}(n)$ are initialized as all-zero vectors while the filters $\mathbf{g}_{2,p}(n)$ are initialized as $\mathbf{g}_{2,p}(n)=[0\ \cdots\ \epsilon\ \cdots\ 0]^T$, $p=1,2,\ldots,P$, where ϵ is set to different values. Fig. 5 shows plots of the improvement of CD, LLR, FWSNR, and PESQ of

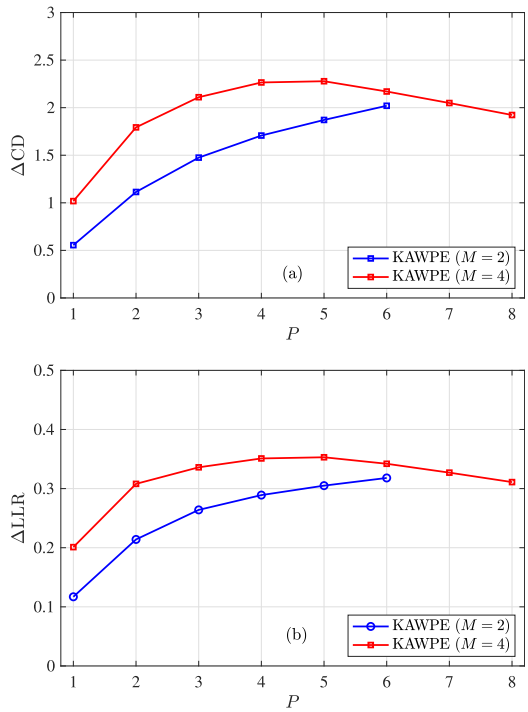


Fig. 3. Improvement of CD and LLR of the KAWPE method as a function of P: (a) ΔCD and (b) ΔLLR. Experimental conditions: $M = 2$, $L_1 = 6$, $L_2 = 6$, and $M = 4$, $L_1 = 8$, $L_2 = 8$.

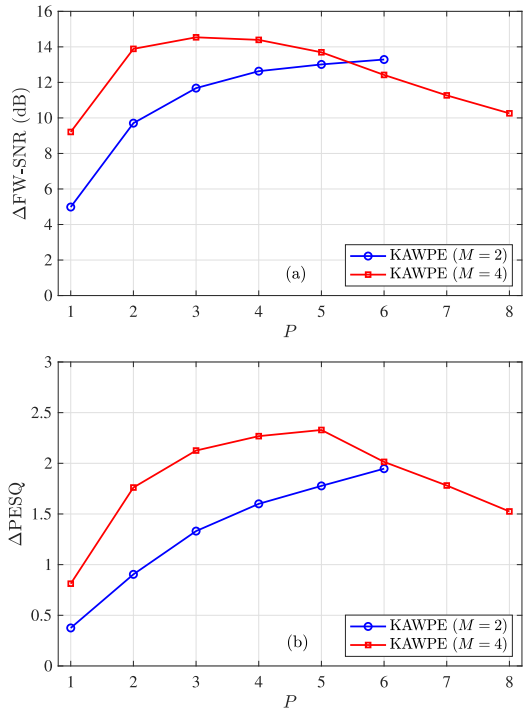


Fig. 4. Improvement of FWSNR and PESQ of the KAWPE method with different values of P: (a) ΔFWSNR and (b) ΔPESQ. Experimental conditions: $M = 2$, $L_1 = 6$, $L_2 = 6$, and $M = 4$, $L_1 = 8$, $L_2 = 8$.

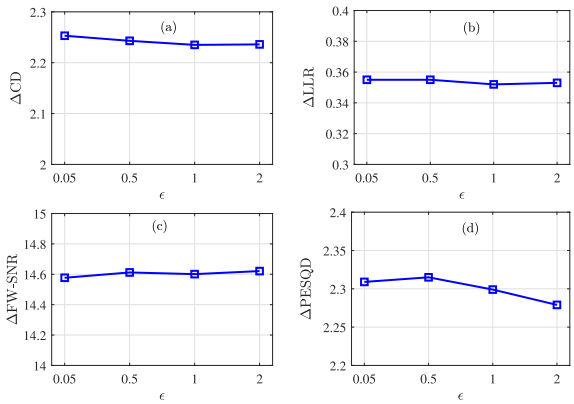


Fig. 5. Performance improvement of the KAWPE method with different italization factor ϵ: (a) ΔCD, (b) ΔLLR, (c) ΔFWSNR, and (d) ΔPESQ. Experimental conditions: $M = 4$, $L_1 = 8$, $L_2 = 8$, and $P = 3$.

KAWPE. It is seen clearly that the performance of KAWPE does not change much with the value of ϵ. We also tried several other initialization methods, and the results are similar, which will not be presented here to make the paper concise.

Now, we study the performance of PTV-KAWPE for speech dereverberation. The conditions are same as those in the previous simulations. We use the first 5-second segment of the observed signals to compute the filters $\mathbf{g}_{2,p}(n)$ and $\mathbf{g}_{1,p}(n)$, $p = 1, 2, \ldots, P$, respectively. Then, we keep the filters $\mathbf{g}_{2,p}$, $p = 1, 2, \ldots, P$, as the time-invariant filters, and only update $\mathbf{g}_{1,p}(n)$. Similar to the previous simulations, we evaluate the performance improvement on a segment basis (with each segment being 2-second long) and with different values of P. Figs. 6 and 7 show plots of the gain in CD, LLR, FWSNR, and PESQ of PTV-KAWPE as a function time. As seen, PTV-KAWPE also achieves significant improvement in CD, LLR, FWSNR, and PESQ, and the dereverberation performance increases with the value of P, which is similar to what was observed in the previous simulations. This validates the feasibility and effectiveness of considering a partially time-varying Kronecker product filtering model for linear prediction-based speech dereverberation.

B. Performance Study With Measured Room Impulse Responses

Now, we study and compare the performance of AWPE and KAWPE with the measured room impulse response. The impulse responses are taken from the Bar-Ilan University (BIU) Acoustic Lab database [48], which consists of multichannel room impulse responses measured in a room of size $6 \times 6 \times 2.4$ m. The reverberation time of the room, i.e., T_{60}, is approximately 610 ms. To measure the impulse responses, a speaker is placed one meter away from a uniform linear microphone array consisting of 8 microphones. The spacing between neighboring microphones is 8 cm. For detailed information about this database, please refer to [48]. In this experiment, we take a set of impulse responses measured with the first four microphones in the array. The experimental configuration is consistent with that in the previous simulations.

The microphone observation signals are generated by convolving the source signal with the corresponding measured acoustic impulse responses. The same source signal used in the

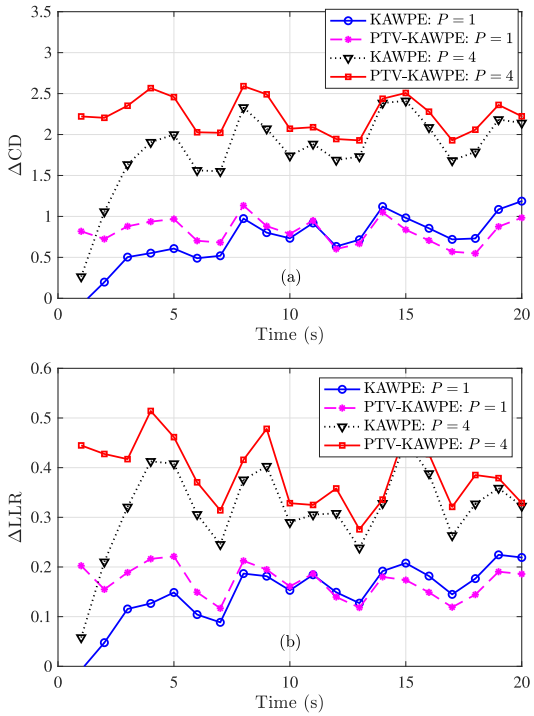


Fig. 6. Improvement of CD and LLR of the KAWPE and PTV-KAWPE methods with different values of P vary with time: (a) ΔCD and (b) ΔLLR. Experimental conditions: $M = 4$, $L_1 = 8$, and $L_2 = 8$.

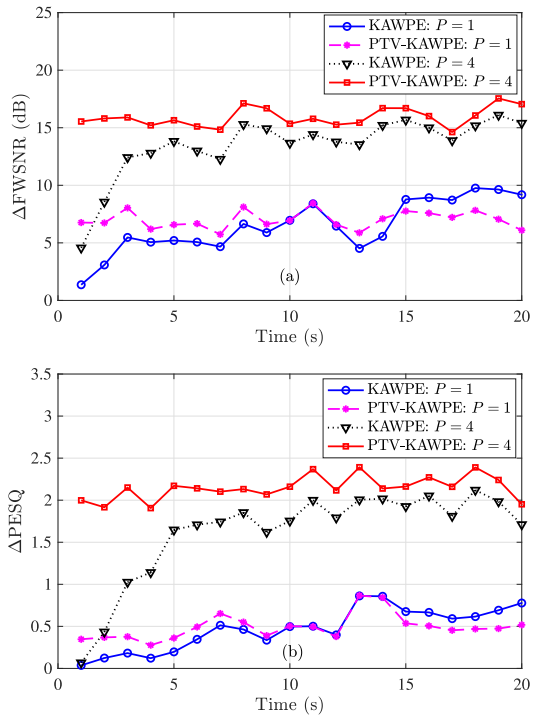


Fig. 7. Improvement of FWSNR and PESQ of the KAWPE and PTV-KAWPE methods with different values of P vary with time: (a) ΔFWSNR and (b) ΔPESQ. Experimental conditions: $M = 4$, $L_1 = 8$, and $L_2 = 8$.

TABLE II
PERFORMANCE OF THE AWPE, KAWPE, AND PTV-KAWPE METHODS. CONDITIONS: $T_{60} \approx 610$ MS AND $M = 4$

L	4	6	8	10	12	14	16
L_M	16	24	32	40	48	56	64
(L_1, L_2)	(4,4)	(6,4)	(8,4)	(8,5)	(8,6)	(8,7)	(8,8)
	ΔCD						
AWPE	0.781	1.156	1.422	1.397	1.253	1.113	0.962
KAWPE							
$P = 1$	0.356	0.457	0.478	0.492	0.480	0.494	0.476
$P = 2$	0.618	0.818	1.041	1.050	1.009	0.978	0.919
$P = 4$	0.914	1.376	1.697	1.699	1.625	1.587	1.548
PTV-KAWPE							
$P = 1$	0.372	0.450	0.512	0.549	0.540	0.584	0.590
$P = 2$	0.608	0.756	0.909	0.960	1.013	1.058	1.039
$P = 4$	0.783	1.143	1.412	1.426	1.439	1.443	1.432
	ΔLLR						
AWPE	0.150	0.192	0.223	0.221	0.205	0.193	0.178
KAWPE							
$P = 1$	0.077	0.096	0.104	0.106	0.102	0.103	0.104
$P = 2$	0.119	0.150	0.179	0.180	0.177	0.170	0.169
$P = 4$	0.162	0.206	0.236	0.232	0.224	0.216	0.216
PTV-KAWPE							
$P = 1$	0.086	0.098	0.108	0.118	0.113	0.122	0.122
$P = 2$	0.122	0.145	0.170	0.175	0.182	0.184	0.188
$P = 4$	0.148	0.191	0.219	0.223	0.224	0.227	0.224
	ΔFWSNR (dB)						
AWPE	7.644	9.751	10.51	10.07	8.847	8.064	7.265
KAWPE							
$P = 1$	3.222	3.989	3.849	3.934	3.480	3.388	3.382
$P = 2$	5.654	6.996	8.450	8.142	7.721	7.299	6.672
$P = 4$	8.459	10.04	10.98	11.10	10.28	10.03	9.745
PTV-KAWPE							
$P = 1$	3.685	4.007	4.400	4.532	4.217	4.557	4.341
$P = 2$	5.744	6.429	7.579	7.447	7.788	7.733	7.499
$P = 4$	7.785	9.285	10.57	10.41	10.38	10.42	10.22
	ΔPESQ						
AWPE	1.076	1.486	1.726	1.699	1.495	1.272	1.177
KAWPE							
$P = 1$	0.419	0.447	0.476	0.508	0.449	0.429	0.427
$P = 2$	0.779	0.914	1.235	1.261	1.205	1.169	0.973
$P = 4$	1.140	1.593	1.905	1.753	1.414	1.238	1.334
PTV-KAWPE							
$P = 1$	0.473	0.501	0.558	0.627	0.576	0.608	0.626
$P = 2$	0.796	0.886	1.096	1.189	1.201	1.263	1.171
$P = 4$	1.050	1.471	1.709	1.744	1.742	1.778	1.718

previous simulations is used here. The parameters for AWPE and KAWPE are set as follows. For AWPE, the recursive factor is set to 0.99, the prediction filter is initialized with an all-zero vector, and the weighted cross-correlation matrix is initialized as an identity matrix. For KAWPE, the parameters are set as: $\alpha_1 = \alpha_2 = 0.99$, $\epsilon = 0.5$, and the value of P varies from 1 to 4. The source signal is 30-s long. Since the PTV-KAWPE algorithm uses the first 5-second signal to estimate the time-invariant filter, the first 5-second output signal is excluded from computing the performance measures for all three methods for a fair comparison. Therefore, the performance measures are computed with a long time average based on 25-s long speech signals. Table II shows the performance gain of AWPE, KAWPE, and PTV-KAWPE with different filter lengths L_M

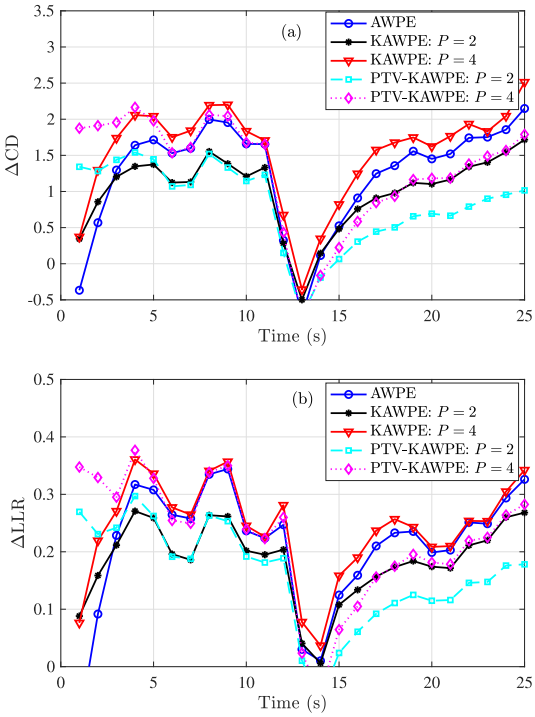


Fig. 8. Improvement of CD and LLR of the AWPE, KAWPE, and PTV-KAWPE methods with different values of P vary with time: (a) ΔCD and (b) ΔLLR. Experimental conditions: $M = 4$, $L_M = 40$, $L_1 = 8$, $L_2 = 5$. The position of the speaker changes at 12.6 seconds.

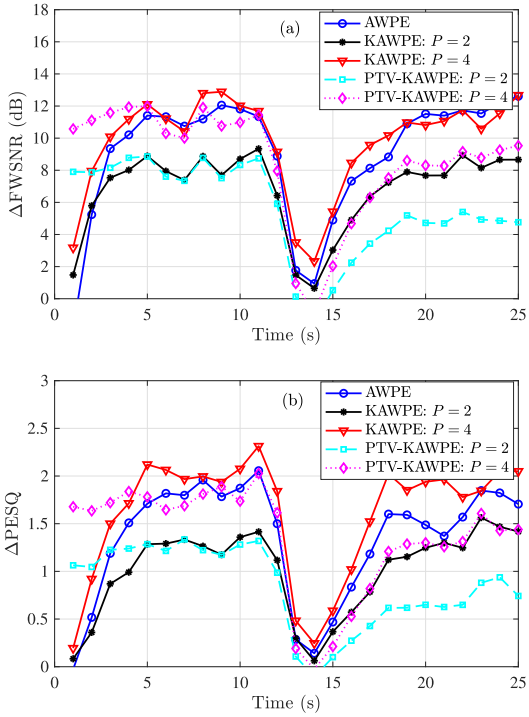


Fig. 9. Improvement of FWSNR and PESQ of the AWPE, KAWPE, and PTV-KAWPE methods with different values of P vary with time: (a) ΔFWSNR and (b) ΔPESQ. Experimental conditions: $M = 4$, $L_M = 40$, $L_1 = 8$, $L_2 = 5$. The position of the speaker changes at 12.6 seconds.

(the values of L_1 and L_2 are also shown in Table II). As seen, all the studied methods improve significantly the performance. Generally, better performance is obtained if a larger value of P is used (note that we should have $P \leq \min(L_1, L_2)$). We see the best performance of the AWPE and KAWPE methods is obtained around $L_M = 40$. In comparison, KAWPE achieves a better overall performance if the order of the Kronecker product is larger than 2. The underlying reason can be explained as follows. If Kronecker product order, i.e., the value of P, is small, KAWPE and PTV-KAWPE have much fewer parameters to estimate, leading to a much lower computational complexity as compared to AWPE. Still, as a result of fewer parameters, the performance suffers from degradation compared to AWPE. As the value of P becomes larger, the performance improvement of KAWPE increases and may even exceed AWPE. Note that in this case, KAWPE may have a similar (or even larger) number of parameters as compared to AWPE, but the matrix dimension related to the adaptive algorithm is smaller and as a result, KAWPE is able to improve the dereverberation performance given the same amount of data for estimation at every time instant.

Now, we consider a moving-source scenario. The impulse responses are again taken from the BIU database, but this time the source position changes suddenly from 0° to 30° at 12.6 seconds. Based on the previous results, we set $L_M = 40$ ($L_1 = 8$, $L_2 = 5$) for all the methods, and the experimental conditions and other parameters are the same as in the previous experiment. Figs. 8 and 9 show plots of the performance gain of AWPE, KAWPE, and PTV-KAWPE as a function of time. As seen at the beginning and also the instant when the speaker's position changes, both AWPE and KAWPE need a similar amount of time to converge. In other words, AWPE and KAWPE have a similar convergence rate. The performance of AWPE is between those of KAWPE with $P = 2$ and $P = 4$. Notice that PTV-KAWPE has a better performance than KAWPE at the beginning since the time-invariant filters are pre-estimated using the first 5-second observation signals. After the source position changes, PTV-KAWPE suffers some performance degradation and its performance is slightly worse than that of KAWPE. The underlying reason for the performance degradation with PTV-KAWPE is that the algorithm still uses the time-invariant filters computed at the first position, which should be different after the source position changes.

In this set of experiments, we consider an environment where both reverberation and background noise exist. The impulse responses are still taken from the BIU database. After convolution, background noise is added to control the SNR to be 20 dB. The background noise consists of white and diffuse noise signals mixed together with an equal level. We set $L_M = 40$ ($L_1 = 8$, $L_2 = 5$) for all three methods, and the other parameters are the same as in the previous experiment. For PTV-KAWPE, the first 5-second segment of the observed signals is used to compute the time-invariant filters, i.e., $\mathbf{g}_{2,p}$, $p = 1, 2, \ldots, P$. Again, the first 5-second segment is not used in evaluation for a fair comparison. So, the performance measures are computed with a long-time average based on 25-s signals. Table III shows the performance

TABLE III
PERFORMANCE OF THE AWPE, KAWPE, AND PTV-KAWPE METHODS IN REVERBERANT AND NOISY ACOUSTIC ENVIRONMENTS. CONDITIONS: $T_{60} \approx 610$ MS, SNR $= 20$ DB, $M = 4$, $L_M = 40$, $L_1 = 8$, $L_2 = 5$

	ΔCD	ΔLLR	ΔFWSNR (dB)	ΔPESQ
AWPE	0.005	0.008	2.468	0.216
KAWPE				
$P = 1$	0.053	0.020	1.345	0.146
$P = 2$	0.088	0.032	2.479	0.260
$P = 4$	0.131	0.035	3.599	0.320
PTV-KAWPE				
$P = 1$	0.031	0.014	1.149	0.127
$P = 2$	0.047	0.022	1.883	0.206
$P = 4$	0.048	0.019	2.587	0.248

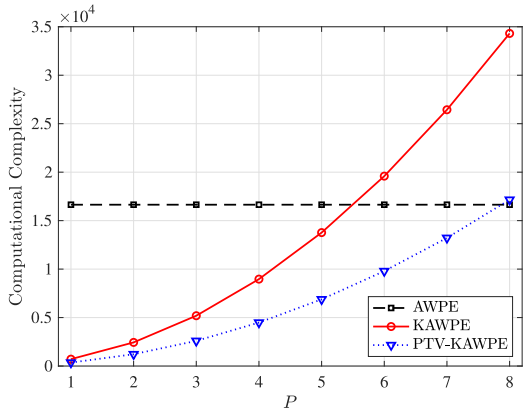


Fig. 10. Computational complexity in terms of complex-valued multiplications of the AWPE, KAWPE, and PTV-KAWPE methods with $L_M = 64$, $L_1 = 8$, $L_2 = 8$.

gain of AWPE, KAWPE, and PTV-KAWPE. As seen, all the three methods still achieved dereverberation in the presence of white and diffuse noise. However, it can be seen that there is less performance improvement with all the three methods as compared to the conditions where there is no additive noise. This shows that additive noise may greatly affect all the WPE-type methods' dereverberation performance. Relatively, KAWPE is slightly less sensitive to noise. How to make the WPE-type of methods robust to additive noise is a topic that has attracted much attention in the field [27], [49], which is beyond the scope of this work.

C. Complexity Study

In this subsection, we compare the computational complexities of the AWPE, KAWPE, and PTV-KAWPE methods. Fig. 10 shows plots of the complexity in terms of complex-valued multiplications of the three methods as a function of P with $L_M = 64$ (the number of microphones is 4 and $L = 16$), $L_1 = 8$, $L_2 = 8$. As seen, KAWPE has lower computational complexity than AWPE as long as the value of P is not too large. In comparison, the complexity of PTV-KAWPE is always lower than those of KAWPE and AWPE. According to the previous experiments, the KAWPE and PTV-KAWPE methods can achieve good dereverberation performance with a small value of P (e.g., when $L_M = 64$, KAWPE and PTV-KAWPE achieved good performance with $P = 3$). In this case, their computational complexity is significantly lower than the AWPE method.

We also performed an informal evaluation by comparing the three algorithms through running the Matlab implementations on a computer with Intel Core i5-4690 3.50-GHz CPU to process 11-second long speech signals recorded with a 4-microphone array. This experiment was repeated 10 times, and the average execution time is computed as the processing time for each algorithm. On average, it takes the AWPE (with $L_M = 64$), KAWPE (with $L_1 = L_2 = 8$ and $P = 1$), and PTV-KAWPE (with $L_1 = L_2 = 8$ and $P = 1$) methods, respectively, 40.5 s, 10.9 s, and 6.8 s to process the 11-second long speech signals. Note that such a method is not very rigorous in terms of complexity evaluation since the results are influenced by several factors such as STFT, data reading, and writing, etc. But the results clearly show that the developed methods are much more efficient than AWPE in computational complexity.

VII. CONCLUSION

Reverberation is one of the principal causes of speech quality and intelligibility degradation, and, as a result, many methods have been developed for dereverberation. Among those, the AWPE method, which attempts to estimate the late reverberation from past observations with a multichannel linear prediction filter, has demonstrated promising potential in real applications. However, the computational complexity of AWPE is high, which restricts its application for real-time applications. This paper introduced a new framework for dereverberation by constructing the linear prediction filter as a Kronecker product of two sets of short filters. We derived the Kronecker product AWPE method for speech dereverberation based on this structure. We then proposed a partially time-varying Kronecker product AWPE for speech dereverberation, which only needs to update part of the Kronecker filters, so the computational efficiency is further improved. Simulation and experimental results showed that both the developed KAWPE and PTV-KAWPE algorithms are computationally much more efficient than AWPE. Yet, they can achieve the same or even better performance than AWPE.

REFERENCES

[1] J. Benesty, J. Chen, and Y. Huang, *Microphone Array Signal Processing*. Berlin, Germany: Springer-Verlag, 2008.
[2] M. Brandstein and D. Ward, *Microphone Arrays: Signal Processing Techniques and Applications*. Berlin, Germany: Springer, 2001.
[3] K. Kinoshita *et al.*, "The REVERB challenge: A common evaluation framework for dereverberation and recognition of reverberant speech," in *Proc. IEEE Workshop Appl. Signal Process. Audio Acoust.*, 2013, pp. 1–4.
[4] D. Schmid, G. Enzner, S. Malik, D. Kolossa, and R. Martin, "Variational bayesian inference for multichannel dereverberation and noise reduction," *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, vol. 22, no. 8, pp. 1320–1335, Aug. 2014.
[5] C. H. Taal, R. C. Hendriks, R. Heusdens, and J. Jensen, "An algorithm for intelligibility prediction of time-frequency weighted noisy speech," *IEEE Trans. Audio, Speech, Lang. Process.*, vol. 19, no. 7, pp. 2125–2136, Sep. 2011.
[6] C. Zheng, X. Li, A. Schwarz, and W. Kellermann, "Statistical analysis and improvement of coherent-to-diffuse power ratio estimators for dereverberation," in *Proc. IEEE Int. Workshop Acoust. Signal Enhancement*, 2016, pp. 1–5.

[7] D. S. Williamson and D. Wang, "Time-frequency masking in the complex domain for speech dereverberation and denoising," *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, vol. 25, no. 7, pp. 1492–1501, Jul. 2017.

[8] O. Schwartz, S. Gannot, and E. A. Habets, "An expectation-maximization algorithm for multimicrophone speech dereverberation and noise reduction with coherence matrix estimation," *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, vol. 24, no. 9, pp. 1495–1510, Sep. 2016.

[9] S. Inoue, H. Kameoka, L. Li, S. Seki, and S. Makino, "Joint separation and dereverberation of reverberant mixtures with multichannel variational autoencoder," in *Proc. IEEE Int. Conf. Acoust. Speech Signal Process.*, 2019, pp. 96–100.

[10] M. Togami, Y. Kawaguchi, R. Takeda, Y. Obuchi, and N. Nukaga, "Optimized speech dereverberation from probabilistic perspective for time varying acoustic transfer function," *IEEE Trans. Audio, Speech, Lang. Process.*, vol. 21, no. 7, pp. 1369–1380, Jul. 2013.

[11] I. Kodrasi and S. Doclo, "Joint dereverberation and noise reduction based on acoustic multichannel equalization," *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, vol. 24, no. 4, pp. 680–693, Apr. 2016.

[12] R. Rashobh, A. Khong, and D. Liu, "Multichannel equalization in the KLT and frequency domains with application to speech dereverberation," *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, vol. 22, no. 3, pp. 634–646, Mar. 2014.

[13] G. Huang, J. Benesty, and J. Chen, "On the design of frequency-invariant beampatterns with uniform circular microphone arrays," *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, vol. 25, no. 5, pp. 1140–1153, May 2017.

[14] G. Huang, J. Benesty, I. Cohen, and J. Chen, "A simple theory and new method of differential beamforming with uniform linear microphone arrays," *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, vol. 28, no. 1, pp. 1079–1093, Mar. 2020.

[15] S. Markovich, S. Gannot, and I. Cohen, "Multichannel eigenspace beamforming in a reverberant noisy environment with multiple interfering speech signals," *IEEE Trans. Audio, Speech, Lang. Process.*, vol. 17, no. 6, pp. 1071–1086, Aug. 2009.

[16] V. M. Tavakoli, J. R. Jensen, M. G. Christensen, and J. Benesty, "Pseudo-coherence-based MVDR beamformer for speech enhancement with ad hoc microphone arrays," in *Proc. IEEE Int. Conf. Acoust. Speech Signal Process.*, 2015, pp. 2659–2663.

[17] G. Huang, J. Chen, and J. Benesty, "Insights into frequency-invariant beamforming with concentric circular microphone arrays," *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, vol. 26, no. 12, pp. 2305–2318, Dec. 2018.

[18] J. S. Erkelens and R. Heusdens, "Correlation-based and model-based blind single-channel late-reverberation suppression in noisy time-varying acoustical environments," *IEEE Trans. Audio, Speech, Lang. Process.*, vol. 18, no. 7, pp. 1746–1765, Sep. 2010.

[19] A. Schwarz and W. Kellermann, "Coherent-to-diffuse power ratio estimation for dereverberation," *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, vol. 23, no. 6, pp. 1006–1018, Jun. 2015.

[20] F. D. Aprilyanti, H. Saruwatari, S. Nakamura, and T. Takatani, "Optimized joint noise suppression and dereverberation based on blind signal extraction for hands-free speech recognition system," in *Proc. 4th Joint Workshop Hands-Free Speech Commun. Microphone Arrays*, 2014, pp. 182–186.

[21] D. Giacobello, M. G. Christensen, M. N. Murthi, S. H. Jensen, and M. Moonen, "Enhancing sparsity in linear prediction of speech by iteratively reweighted 1-norm minimization," in *Proc. IEEE Int. Conf. Acoust. Speech Signal Process.*, 2010, pp. 4650–4653.

[22] V. M. Tavakoli, J. R. Jensen, M. G. Christensen, and J. Benesty, "A framework for speech enhancement with ad hoc microphone arrays," *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, vol. 24, no. 6, pp. 1038–1051, Jun. 2016.

[23] T. Nakatani, T. Yoshioka, K. Kinoshita, M. Miyoshi, and B.-H. Juang, "Blind speech dereverberation with multichannel linear prediction based on short time fourier transform representation," in *Proc. IEEE Int. Conf. Acoust. Speech Signal Process.*, 2008, pp. 85–88.

[24] T. Yoshioka, T. Nakatani, and M. Miyoshi, "Integrated speech enhancement method using noise suppression and dereverberation," *IEEE Trans. Audio, Speech, Lang. Process.*, vol. 17, no. 2, pp. 231–246, Feb. 2009.

[25] R. Ikeshita, N. Kamo, and T. Nakatani, "Blind signal dereverberation based on mixture of weighted prediction error models," *IEEE Signal Process. Lett.*, vol. 28, pp. 399–403, 2021.

[26] T. Nakatani and K. Kinoshita, "A unified convolutional beamformer for simultaneous denoising and dereverberation," *IEEE Signal Process. Lett.*, vol. 26, no. 6, pp. 903–907, Jun. 2019.

[27] T. Nakatani and K. Kinoshita, "Maximum likelihood convolutional beamformer for simultaneous denoising and dereverberation," in *Proc. 27th Eur. Signal Process. Conf.*, 2019, pp. 1–5.

[28] S. Song, L. Cheng, S. Luan, D. Yao, J. Li, and Y. Yan, "An integrated multichannel approach for joint noise reduction and dereverberation," *Appl. Acoust.*, vol. 171, 2021, Art. no. 107526.

[29] W. Zhang *et al.*, "End-to-end dereverberation, beamforming, and speech recognition with improved numerical stability and advanced frontend," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process.*, 2021, pp. 6898–6902.

[30] S. Hashemgeloogerdi and S. Braun, "Joint beamforming and reverberation cancellation using a constrained Kalman filter with multichannel linear prediction," in *Proc. IEEE Int. Conf. Acoust. Speech Signal Process.*, 2020, pp. 481–485.

[31] 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, Sep. 2010.

[32] T. Yoshioka, H. Tachibana, T. Nakatani, and M. Miyoshi, "Adaptive dereverberation of speech signals with speaker-position change detection," in *Proc. IEEE Int. Conf. Acoust. Speech Signal Process.*, 2009, pp. 3733–3736.

[33] T. Xiang, J. Lu, and K. Chen, "Multichannel adaptive dereverberation robust to abrupt change of target speaker position," *J. Acoust. Soc. Amer.*, vol. 145, no. 3, pp. EL250–EL256, 2019.

[34] J. Heymann, L. Drude, R. Haeb-Umbach, K. Kinoshita, and T. Nakatani, "Frame-online DNN-WPE dereverberation," in *Proc. IEEE Int. Workshop Acoust. Signal Enhancement*, 2018, pp. 466–470.

[35] W. Yang, G. Huang, J. Chen, J. Benesty, I. Cohen, and W. Kellermann, "Robust dereverberation with Kronecker product based multichannel linear prediction," *IEEE Signal Process. Lett.*, vol. 28, pp. 101–105, 2021.

[36] Y. I. Abramovich, G. J. Frazer, and B. A. Johnson, "Iterative adaptive Kronecker MIMO radar beamformer: Description and convergence analysis," *IEEE Trans. Signal Process.*, vol. 58, no. 7, pp. 3681–3691, Jul. 2010.

[37] G. Huang, I. Cohen, J. Benesty, and J. Chen, "Kronecker product beamforming with multiple differential microphone arrays," in *Proc. IEEE 11th Sensor Array Multichannel Signal Process. Workshop*, 2020, pp. 1–5.

[38] J. Benesty, I. Cohen, and J. Chen, *Array Processing-Kronecker Product Beamforming*, vol. 18. Berlin, Germany: Springer, 2019.

[39] G. Huang, J. Benesty, J. Chen, and I. Cohen, "Robust and steerable Kronecker product differential beamforming with rectangular microphone arrays," in *Proc. IEEE Int. Conf. Acoust. Speech Signal Process.*, 2020, pp. 211–215.

[40] J. Benesty, C. Paleologu, and S. Ciochină, "On the identification of bilinear forms with the Wiener filter," *IEEE Signal Process. Lett.*, vol. 24, no. 5, pp. 653–657, May 2017.

[41] C. Paleologu, J. Benesty, and S. Ciochina, "Linear system identification based on a Kronecker product decomposition," *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, vol. 26, no. 10, pp. 1793–1808, Oct. 2018.

[42] J. Benesty, C. Paleologu, L.-M. Dogariu, and S. Ciochina, "Identification of linear and bilinear systems: A unified study," *Electronics*, vol. 10, no. 15, Jul. 2021, Art. no. 33.

[43] D. A. Harville, *Matrix Algebra From a Statistician's Perspective*. New York, NY, USA: Taylor Francis Group, 1998.

[44] C. Elisei-Iliescu, C. Paleologu, J. Benesty, C. Stanciu, C. Anghel, and S. Ciochină, "Recursive least-squares algorithms for the identification of low-rank systems," *IEEE Trans. Audio, Speech, Lang. Process.*, vol. 27, no. 5, pp. 903–918, May 2019.

[45] J. Allen and D. Berkley, "Image method for efficiently simulating small-room acoustics," *J. Acoust. Soc. Amer.*, vol. 65, no. 4, pp. 943–950, 1979.

[46] K. Kinoshita *et al.*, "A summary of the REVERB challenge: State-of-the-art and remaining challenges in reverberant speech processing research," *EURASIP J. Adv. Signal Process.*, vol. 2016, no. 1, 2016, Art. no. 7.

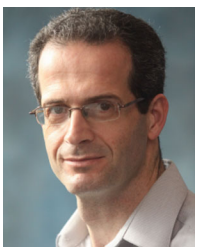
[47] Y. Hu and P. C. Loizou, "Evaluation of objective quality measures for speech enhancement," *IEEE Trans. Audio, Speech, Lang. Process.*, vol. 16, no. 1, pp. 229–238, Jan. 2008.

[48] E. Hadad, F. Heese, P. Vary, and S. Gannot, "Multichannel audio database in various acoustic environments," in *Proc. IEEE Int. Workshop Acoust. Signal Enhancement*, 2014, pp. 313–317.

[49] W. Yang, G. Huang, W. Zhang, J. Chen, and J. Benesty, "Dereverberation with differential microphone arrays and the weighted-prediction-error method," in *Proc. IEEE Int. Workshop Acoust. Signal Enhancement*, 2018, pp. 376–380.

Gongping Huang (Member, IEEE) received the bachelor's degree in electronics and information engineering and the Ph.D. degree in information and communication engineering from Northwestern Polytechnical University, Xian, China, in 2012 and 2019, respectively. He is currently a Humboldt Research Fellow with the Chair of LMS, University of Erlangen-Nuremberg, Erlangen, Germany. He was an Andrew and Erna Finci Viterbi Postdoctoral Research Fellow with the Technion–Israel Institute of Technology, Haifa, Israel. From 2015 to 2017, he was a Visiting Researcher with the University of Quebec, INRS-EMT, Montreal, QC, Canada. His research interests include microphone arrays, acoustic signal processing, and speech enhancement. He was the recipient of the Humboldt Research Fellowship for Postdoctoral Researchers (2021), the Andrew and Erna Finci Viterbi Post-Doctoral Fellowship's Award (2019), the Best Ph.D. Thesis Award of Chinese Institute of Electronics (2021), and the Best Ph.D. Thesis Award of Shannxi Province (2021). He is an active Reviewer for many scientific journals and international conferences include IEEE TASL, IEEE TSP, IEEE TMM, IEEE SPL, IEEE TVT, IEEE TMECH, JASA, Speech Communications, Signal Processing, ICASSP, IWAENC, INTERSPEECH, EUSIPCO.

Jacob Benesty received the master's degree in microwaves from Pierre & Marie Curie University, Paris, France, in 1987, and the Ph.D. degree in control and signal processing from Orsay University, Bures-sur-Yvette, France, in April 1991. During the Ph.D. (from November 1989 to April 1991), he worked on adaptive filters and fast algorithms with the Centre National d'Etudes des Telecommunications (CNET), Paris, France. From January 1994 to July 1995, he was with Telecom Paris University, Paris, France, on multichannel adaptive filters and acoustic echo cancellation. From October 1995 to May 2003, he was first a Consultant and then a Member of the Technical Staff at Bell Laboratories, Murray Hill, NJ, USA. In May 2003, he joined the University of Quebec, INRS-EMT, in Montreal, Quebec, Canada, as a Professor. He is also an Adjunct Professor with Aalborg University, Aalborg, Denmark, and a Guest Professor with Northwestern Polytechnical University, Xi'an, China. He has coauthored and co-edited and coauthored numerous books in the area of acoustic signal processing. His research interests include signal processing, acoustic signal processing, and multimedia communications. He is the Inventor of many important technologies. In particular, he was the Lead Researcher with Bell Labs who conceived and designed the world-first real-time hands-free full-duplex stereophonic teleconferencing system. Also, he conceived and designed the world-first PC-based multi-party hands-free full-duplex stereo conferencing system over IP networks. He is the Editor of the book series Springer Topics in Signal Processing. He was the General Chair and Technical Chair of many international conferences and a Member of several IEEE technical committees. Four of his journal papers were awarded by the IEEE Signal processing Society and in 2010 he was the recipient of the Gheorghe Cartianu Award from the Romanian Academy.



Israel Cohen (Fellow, IEEE) received the B.Sc. (*Summa Cum Laude*), M.Sc., and Ph.D. degrees in electrical engineering from the Technion – Israel Institute of Technology, Haifa, Israel, in 1990, 1993 and 1998, respectively. He is currently a Professor of electrical engineering with the the Technion – Israel Institute of Technology. He is also a Visiting Professor with Northwestern Polytechnical University, Xi'an, China. From 1990 to 1998, he was a Research Scientist with RAFAEL Research Laboratories, Haifa, Israel Ministry of Defense. From 1998 to 2001, he was a Postdoctoral Research Associate with Computer Science Department, Yale University, New Haven, CT, USA. In 2001, he joined the Electrical Engineering Department of the Technion. His research interests include array processing, statistical signal processing, analysis and modeling of acoustic signals, speech enhancement, noise estimation, microphone arrays, source localization, blind source separation, system identification and adaptive filtering. He is a co-editor of the Multichannel Speech Processing Section of the *Springer Handbook of Speech Processing* (Springer, 2008), and a coauthor of *Fundamentals of Signal Enhancement and Array Signal Processing* (Wiley-IEEE Press, 2018). Dr. Cohen was the recipient of the Norman Seiden Prize for Academic Excellence (2017), the SPS Signal Processing Letters Best Paper Award (2014), the Alexander Goldberg Prize for Excellence in Research (2010), and the Muriel and David Jacknow Award for Excellence in Teaching (2009). He is an Associate Member of the IEEE Audio and Acoustic Signal Processing Technical Committee, and as Distinguished Lecturer of the IEEE Signal Processing Society. He was an Associate Editor of the IEEE TRANSACTIONS ON AUDIO, SPEECH, AND LANGUAGE PROCESSING and IEEE SIGNAL PROCESSING LETTERS, and as Member of the IEEE Audio and Acoustic Signal Processing Technical Committee and the IEEE Speech and Language Processing Technical Committee.

Jingdong Chen (Fellow, IEEE) received the Ph.D. degree in pattern recognition and intelligence control from the Chinese Academy of Sciences, Beijing, China, in 1998. From 1998 to 1999, he was with ATR Interpreting Telecommunications Research Laboratories, Kyoto, Japan, where he conducted research on speech synthesis, speech analysis, as well as objective measurements for evaluating speech synthesis. Then he joined the Griffith University, Brisbane, Australia, where he engaged in research on robust speech recognition and signal processing. From 2000 to 2001, he was with ATR Spoken Language Translation Research Laboratories on robust speech recognition and speech enhancement. From 2001 to 2009, he was a Member of Technical Staff with Bell Laboratories, Murray Hill, New Jersey, working on acoustic signal processing for telecommunications. He subsequently joined WeVoice Inc., New Jersey, as the Chief Scientist. He is currently a Professor with Northwestern Polytechnical University, Xi'an, China. His research interests include array signal processing, adaptive signal processing, speech enhancement, adaptive noise/echo control, signal separation, speech communication, and artificial intelligence. He was an Associate Editor of the IEEE TRANSACTIONS ON AUDIO, SPEECH, AND LANGUAGE PROCESSING from 2008 to 2014 and as a Technical Committee (TC) Member of the IEEE Signal Processing Society (SPS) TC on Audio and Electroacoustics from 2007 to 2009. He is currently serving as the Chair of IEEE Xi'an Section and a member of the IEEE SPS TC on Audio and Acoustic Signal Processing. He was the General Co-Chair of ACM WUWNET 2018 and IWAENC 2016, the Technical Program Chair of IEEE TENCON 2013, a Technical Program Co-Chair of IEEE WASPAA 2009, IEEE ChinaSIP 2014, IEEE ICSPCC 2014, and IEEE ICSPCC 2015, and helped organize many other conferences. He co-authored 12 monograph books including *Array Processing–Kronecker Product Beamforming*, (Springer, 2019), *Fundamentals of Signal Enhancement and Array Signal Processing*, (Wiley, 2018), *Fundamentals of Differential Beamforming*, (Springer, 2016), *Design of Circular Differential Microphone Arrays* (Springer, 2015), *Noise Reduction in Speech Processing* (Springer, 2009), *Microphone Array Signal Processing* (Springer, 2008), and *Acoustic MIMO Signal Processing* (Springer, 2006). Dr. Chen was the recipient of the 2008 Best Paper Award from the IEEE Signal Processing Society (with Benesty, Huang, and Doclo), the Best Paper Award from the IEEE Workshop on Applications of Signal Processing to Audio and Acoustics in 2011 (with Benesty), the Bell Labs Role Model Teamwork Award twice, respectively, in 2009 and 2007, the NASA Tech Brief Award twice, respectively, in 2010 and 2009, and the Young Author Best Paper Award from the 5th National Conference on Man-Machine Speech Communications in 1998. He is a co-author of a paper for which C. Pan was the recipient of the IEEE R10 (Asia-Pacific Region) Distinguished Student Paper Award (First Prize) in 2016. He was also the recipient of the Japan Trust International Research Grant from the Japan Key Technology Center in 1998 and the Distinguished Young Scientists Fund from the National Natural Science Foundation of China in 2014.