Multichannel Iterative Noise Reduction Filters in the Short-Time-Fourier-Transform Domain Based on Kronecker Product Decomposition

Xianghui Wang, Jie Chen, *Senior Member, IEEE*, Xiaoyi Chen, Jing Guo, and Qian Xiang

***Abstract*—In this paper, the design of multichannel noise reduction filters in the short-time-Fourier-transform (STFT) domain is addressed. By investigating the structure of the linear filter, a set of multichannel iterative noise reduction filters are developed based on the Kronecker product decomposition. Instead of computing a long noise reduction filter, we compute three much shorter sub-filters which are separately applied in the spatial, temporal and frequency dimensions. Consequently, compared with the traditional multichannel STFT-domain noise reduction filters, the proposed approaches have two advantages: 1) significantly lower computational complexity; 2) less past observations are needed to construct the iterative filters, which leads to better tracking ability for the temporal/spatial signal nonstationarity. Experimental results demonstrate the advantages of the developed iterative filters over the traditional ones.**

***Index Terms*—Noise reduction, iterative filter, Kronecker product decomposition, STFT domain, multichannel.**

I. Introduction

NOISE reduction has long been an important research topic in the field of speech signal processing. It has attracted a significant amount of attention over the last few decades [1]–[9]. Compared to using a single microphone, it has been proved in both theory and practice that using multiple microphones or microphone arrays is beneficial for achieving enhanced performance [10], [11], leading to the research topic of multichannel noise reduction.

Multichannel noise reduction is typically achieved by applying a spatial-temporal filter/beamformer to the signals received by an array of microphones. Researchers aim to design a proper spatial-temporal filter/beamformer so that the noise is significantly reduced while the signal of interest is preserved with little or even no distortion when processing noisy signals. At the beginning, peoples focused on recovering the original source signal by designing different kinds of beamformers, such as delay-and-sum beamformer [12], [13], minimum variance distortionless response (MVDR) beamformer [14], linearly constrained minimum variance (LCMV) beamformer [15] and generalized sidelobe canceller (GSC) [16], which are extensively studied and applied in narrowband signal processing. These techniques are designed to conduct both dereverberation and noise reduction at the same time. However, the reverberant components of the source signal usually cause serious speech distortion in highly reverberant environments. In addition, speech dereverberation alone is a challenging task, no mention of simultaneously considering both dereverberation and noise reduction [17]. In order to overcome these drawbacks, algorithms are proposed to recover the speech components observed at a reference microphone [18]–[20]. These methods usually achieve better noise reduction performance in practice. Then, many multichannel noise reduction methods have been developed based on this framework. Representative ones include the multichannel Wiener filter [21]–[23], MVDR filter [24]–[26], LCMV filter [24], [27], GSC [18], [19], [28], and spatial-temporal prediction method [20], [21], etc. While those algorithms are successful to certain degree in dealing with the noise reduction problem, how to make multichannel noise reduction filters effective and efficient is still an open problem, especially when both the informations of inter-frame and inter-band correlation of the speech signal are taken into account [29]–[34]. So, further research attention is indispensable.

This paper is devoted to the problem of multichannel noise reduction in the STFT domain. It is well known that, in the STFT domain, considering the inter-frame and inter-band information of the speech signal improves the noise reduction performance [29]–[31]. However, long (high-order) filters are needed in this case, which introduces the following two problems. First, the computational complexity increases dramatically with the increase of the filter length. Second, longer filters require more past data for a confident estimation of signal covariance matrices, which degrades the algorithm tracking ability for the temporal/spatial signal nonstationarity. In order to alleviate these issues related with the filter length, we resort to the Kronecker product decomposition and nearest Kronecker product approximation.

Manuscript received December 12, 2020; revised May 13, 2021; accepted June 11, 2021. Date of publication June 28, 2021; date of current version August 21, 2021. This work was supported in part by the National Science Foundation of China (NSFC) under Grant 61671382. The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Andy W H Khong. *(Corresponding author: Xianghui Wang.)*

Xianghui Wang, Xiaoyi Chen, Jing Guo, and Qian Xiang are with the Center of Intelligent Acoustics and Immersive Communications of the School of Electronic Information and Artificial Intelligence, Shaanxi University of Science and Technology, Xi'an 710021, China, and also with the Shaanxi Joint Laboratory of Artificial Intelligence, Shaanxi University of Science and Technology, Xi'an 710021, China (e-mail: wangxh1985@126.com; chenxiaoyi@sust.edu.cn; snowy0821@163.com; xiangqian@fynu.edu.cn).

Jie Chen is with the Center of Intelligent Acoustics and Immersive Communications, and the School of Marine Science and Technology, Northwestern Polytechnical University, Xi'an 710072, China (e-mail: dr.jie.chen@ieee.org).

Digital Object Identifier 10.1109/TASLP.2021.3092825

The Kronecker product decomposition and nearest Kronecker product approximation were exploited in many areas, such as data mining [35]–[37], machine learning [38], [39], source separation [40], system identification [41]–[46], and beamforming [47]–[49], etc. Most of these approaches are related to the identification of bilinear/trilinear forms, based on the tensor decomposition and modeling. By exploiting certain properties, e.g., rank-deficient or replicability of the data matrix/tensor, impulse response and sensor array, many high-dimensional problems can be transformed into low-dimensional ones based on the Kronecker product decomposition.

In this work, we propose a noise reduction framework that approximately separates a long filter into three sub-filters that explicitly operate in the spatial, temporal and frequency dimensions, respectively. These sub-filters possess clear physical interpretation, and the dimensions of these sub-filters are much shorter than the original long filter. Less past samples are then required to estimate the signal covariance matrices since the sub-filters consist of much less number of coefficients, which means better ability in tracking the temporal/spatial signal non-stationarity. Under this developed framework, we deduce a set of iterative multichannel noise reduction filters, including the iterative Wiener, the iterative MVDR and the iterative tradeoff filters. Simulation results illustrate that under proper settings, the proposed iterative filters can achieve comparable or even better noise reduction performance with significantly lower computational complexity compared with their prototype with full filter representation.

The remainder of this manuscript is organized as follows. Section II presents the signal model and problem formulation. Some performance measures are defined in Section III. A set of iterative noise reduction filters are developed in Section IV. Section V analyzes the computational complexity of the iterative Wiener filter as an example. Simulations are conducted in Section VI and, finally, conclusions are drawn in Section VII.

II. Signal Model and Problem Formulation

Within the scenario of multichannel noise reduction [2], [4], [21], the following signal model is considered in this work:

$$y_m(t) = x_m(t) + v_m(t),\ m = 1, 2, \ldots, M, \tag{1}$$

where t denotes the discrete time index, $x_m(t)$ and $v_m(t)$ are respectively the speech signal and additive noise captured by the mth microphone, and assumed to be uncorrelated with each other, $y_m(t)$ is the noisy signal at the mth microphone, and M is the number of microphones. We consider all of the signals to be real, broadband, and zero mean. Without loss of generality, we choose $x_1(t)$ as the desired signal, which needs to be recovered from the observations.

Using the STFT, (1) can be rewritten in the time-frequency domain as [4], [29]

$$Y_m(k,n) = X_m(k,n) + V_m(k,n),\ m = 1, 2, \ldots, M, \tag{2}$$

where the zero-mean complex random variables $Y_m(k,n)$, $X_m(k,n)$, and $V_m(k,n)$ are the STFTs of $y_m(t)$, $x_m(t)$, and $v_m(t)$, respectively, at frequency bin $k \in \{0, 1, \ldots, K-1\}$ and time frame n. Accordingly, the sample $X_1(k,n)$ is considered as the desired signal.

As pointed out in [29]–[31], in speech noise reduction, due to the self-correlation of speech signal and the short length of signal frame, we generally have $\phi_{X_m}(k,k,'n) = E[X_m(k,n)X_m(k,'n)] \neq 0$ and $\phi_{X_m}(k,n,n') = E[X_m(k,n)X_m(k,n')] \neq 0$, when the distance between k/n and k'/n' is not too far, where $E[\cdot]$ denotes mathematical expectation. In order to make full use of the available inter-frame and inter-band information, the speech signal is usually organized as

$$\underline{\mathbf{x}}(k,n) = \left[\ \underline{\mathbf{x}}_1^T(k,n)\ \ \underline{\mathbf{x}}_2^T(k,n)\ \ \cdots\ \ \underline{\mathbf{x}}_M^T(k,n)\ \right]^T, \tag{3}$$

where

$$\underline{\mathbf{x}}_m(k,n) = \left[\ \mathbf{x}_m^T(k,n)\ \ \mathbf{x}_m^T(k,n-1)\ \cdots\ \mathbf{x}_m^T(k,n-N+1)\ \right]^T, \tag{4}$$

$$\mathbf{x}_m(k,n) = \left[\ X_m(k-K_k^-,n)\ \cdots\ X_m(k-1,n)\ X_m(k,n)\ X_m(k+1,n)\ \cdots\ X_m(k+K_k^+,n)\ \right]^T, \tag{5}$$

and $(\cdot)^T$ is the transpose operator. The length of the vector $\underline{\mathbf{x}}(k,n)$, $\underline{\mathbf{x}}_m(k,n)$ and $\mathbf{x}_m(k,n)$ are, respectively, MNL, NL and L, where $L = K_k^- + K_k^+ + 1 \leq K$. Then, the signal model can be written as

$$\underline{\mathbf{y}}(k,n) = \underline{\mathbf{x}}(k,n) + \underline{\mathbf{v}}(k,n), \tag{6}$$

where $\underline{\mathbf{y}}(k,n)$ and $\underline{\mathbf{v}}(k,n)$ are defined similarly to $\underline{\mathbf{x}}(k,n)$.

Typically, the desired signal $X_1(k,n)$ is estimated by passing the noisy signal vector, $\underline{\mathbf{y}}(k,n)$, through a linear filter, $\underline{\mathbf{h}}(k,n)$, of length MNL, i.e.,

$$Z(k,n) = \underline{\mathbf{h}}^H(k,n)\underline{\mathbf{y}}(k,n), \tag{7}$$

where $(\cdot)^H$ is the conjugate-transpose operator. Accordingly, in the traditional noise reduction algorithms, we usually need to compute a noise reduction filter, $\underline{\mathbf{h}}(k,n)$, with MNL coefficients.

In order to deduce the proposed noise reduction algorithms, let us form a $M \times N \times L$ noisy signal tensor $\boldsymbol{\mathcal{Y}}(k,n)$, where

$$\boldsymbol{\mathcal{Y}}_{:i_n i_l}(k,n) = \left[\ Y_1(k-i_l+1,n-i_n+1)\ Y_2(k-i_l+1,n-i_n+1)\ \cdots\ Y_M(k-i_l+1,n-i_n+1)\ \right]^T, \tag{8}$$

$$\boldsymbol{\mathcal{Y}}_{i_m:i_l}(k,n) = \left[\ Y_{i_m}(k-i_l+1,n)\ Y_{i_m}(k-i_l+1,n-1)\ \cdots\ Y_{i_m}(k-i_l+1,n-N+1)\ \right]^T, \tag{9}$$

$$\boldsymbol{\mathcal{Y}}_{i_m i_n:}(k,n) = \left[\ Y_{i_m}(k-K_k^-,n-i_n+1)\ \cdots\ Y_{i_m}(k,n-i_n+1)\ \cdots\ Y_{i_m}(k+K_k^+,n-i_n+1)\ \right]^T, \tag{10}$$

with $1 \leq i_m \leq M$, $1 \leq i_n \leq N$ and $1 - K_k^+ \leq i_l \leq 1 + K_k^-$. Note that $\underline{\mathbf{y}}(k,n) = [\mathrm{vec}^T[(\boldsymbol{\mathcal{Y}}_{1::})^T(k,n)]\mathrm{vec}^T[(\boldsymbol{\mathcal{Y}}_{2::})^T(k,n)]$

$\cdots \mathrm{vec}^T[(\boldsymbol{\mathcal{Y}}_{M::})^T(k,n)]]^T$, where $\mathrm{vec}[\cdot]$ is the vectorization operation of a matrix or tensor. Now, we need a filtering tensor $\boldsymbol{\mathcal{H}}(k,n)$ of size $M \times N \times L$ to perform noise reduction, i.e.,

$$Z(k,n) = \langle \boldsymbol{\mathcal{H}}(k,n),\ \boldsymbol{\mathcal{Y}}(k,n) \rangle, \tag{11}$$

where

$$\boldsymbol{\mathcal{Y}}(k,n) = \boldsymbol{\mathcal{X}}(k,n) + \boldsymbol{\mathcal{V}}(k,n), \tag{12}$$

$\boldsymbol{\mathcal{X}}(k,n)$ and $\boldsymbol{\mathcal{V}}(k,n)$ are, respectively, the clean speech and noise signal tensor of size $M \times N \times L$ defined similarly to $\boldsymbol{\mathcal{Y}}(k,n)$,

$$\underline{\mathbf{h}}(k,n) = \Big[\ \mathrm{vec}^T\left[(\boldsymbol{\mathcal{H}}_{1::})^T(k,n)\right]\ \ \mathrm{vec}^T\left[(\boldsymbol{\mathcal{H}}_{2::})^T(k,n)\right] \cdots \mathrm{vec}^T\left[(\boldsymbol{\mathcal{H}}_{M::})^T(k,n)\right]\ \Big]^T, \tag{13}$$

and $< \boldsymbol{\mathcal{A}},\ \boldsymbol{\mathcal{B}} >$ denotes inner product of the tensors $\boldsymbol{\mathcal{A}}$ and $\boldsymbol{\mathcal{B}}$. For ease of exposition, we will drop k and n from the notation from now on wherever there is no ambiguity.

As we know, the filtering tensor $\boldsymbol{\mathcal{H}}$ can be decomposed as [36], [37], [50]

$$\boldsymbol{\mathcal{H}} = \sum_{r=1}^{R} \mathbf{h}_{\mathrm{S},r} \circ \mathbf{h}_{\mathrm{T},r} \circ \mathbf{h}_{\mathrm{F},r}, \tag{14}$$

where the length of $\mathbf{h}_{\mathrm{S},r}$, $\mathbf{h}_{\mathrm{T},r}$ and $\mathbf{h}_{\mathrm{F},r}$ are, respectively, M, N and L, R is the rank of $\boldsymbol{\mathcal{H}}$, the symbols 'S,' 'T' and 'F' denote, respectively, spatial, temporal and frequency, and $\circ$ denotes vector outer product. Note that the scaling ambiguity caused by

$$\mathbf{h}_{\mathrm{S},r} \circ \mathbf{h}_{\mathrm{T},r} \circ \mathbf{h}_{\mathrm{F},r} = (\alpha_r \mathbf{h}_{\mathrm{S},r}) \circ (\beta_r \mathbf{h}_{\mathrm{T},r}) \circ (\gamma_r \mathbf{h}_{\mathrm{F},r}) \tag{15}$$

has no effect on the filtering tensor $\boldsymbol{\mathcal{H}}$, where $\alpha_r \beta_r \gamma_r = 1,\ r = 1,\ 2,\ \ldots,\ R$.

It is obvious that the fibers of mode-q ($q = 1,\ 2,\ 3$) of the noisy signal tensor $\boldsymbol{\mathcal{Y}}$ are strongly correlated with each other, and so are the fibers of mode-q ($q = 1,\ 2,\ 3$) of the filtering tensor $\boldsymbol{\mathcal{H}}$. Thus, in general, $\boldsymbol{\mathcal{H}}$ can be well approximated by a small number of rank-one tensors, i.e.,

$$\boldsymbol{\mathcal{H}} \approx \sum_{p=1}^{P} \mathbf{h}_{\mathrm{S},p} \circ \mathbf{h}_{\mathrm{T},p} \circ \mathbf{h}_{\mathrm{F},p} \tag{16}$$

where $P \leq R$. Accordingly, the filter $\underline{\mathbf{h}}$ can be approximated as

$$\begin{aligned} \underline{\mathbf{h}} \approx \underline{\mathbf{h}}_P &\triangleq \mathrm{vec}\left(\sum_{p=1}^{P} \mathbf{h}_{\mathrm{S},p} \circ \mathbf{h}_{\mathrm{T},p} \circ \mathbf{h}_{\mathrm{F},p}\right) \\ &= \sum_{p=1}^{P} \mathbf{h}_{\mathrm{F},p} \otimes \mathbf{h}_{\mathrm{T},p} \otimes \mathbf{h}_{\mathrm{S},p}, \end{aligned} \tag{17}$$

where $\otimes$ denotes the Kronecker product. Note that when $P = R$, $\underline{\mathbf{h}}_P = \underline{\mathbf{h}}$.

Using the relations:

$$\begin{aligned} \mathbf{h}_{\mathrm{F},p} \otimes \mathbf{h}_{\mathrm{T},p} \otimes \mathbf{h}_{\mathrm{S},p} &= (\mathbf{h}_{\mathrm{F},p} \otimes \mathbf{h}_{\mathrm{T},p} \otimes \mathbf{I}_M)\, \mathbf{h}_{\mathrm{S},p} \\ &= (\mathbf{h}_{\mathrm{F},p} \otimes \mathbf{I}_N \otimes \mathbf{h}_{\mathrm{S},p})\, \mathbf{h}_{\mathrm{T},p} \\ &= (\mathbf{I}_L \otimes \mathbf{h}_{\mathrm{T},p} \otimes \mathbf{h}_{\mathrm{S},p})\, \mathbf{h}_{\mathrm{F},p}, \end{aligned} \tag{18}$$

we can write $\underline{\mathbf{h}}_P$ in the following equivalent forms:

$$\underline{\mathbf{h}}_P = \sum_{p=1}^{P} \mathbf{H}_{\mathrm{TF},p} \mathbf{h}_{\mathrm{S},p} = \sum_{p=1}^{P} \mathbf{H}_{\mathrm{SF},p} \mathbf{h}_{\mathrm{T},p} = \sum_{p=1}^{P} \mathbf{H}_{\mathrm{ST},p} \mathbf{h}_{\mathrm{F},p}, \tag{19}$$

where $\mathbf{H}_{\mathrm{TF},p} = \mathbf{h}_{\mathrm{F},p} \otimes \mathbf{h}_{\mathrm{T},p} \otimes \mathbf{I}_M$ is a matrix of size $MNL \times M$, $\mathbf{H}_{\mathrm{SF},p} = \mathbf{h}_{\mathrm{F},p} \otimes \mathbf{I}_N \otimes \mathbf{h}_{\mathrm{S},p}$ is a matrix of size $MNL \times N$, $\mathbf{H}_{\mathrm{ST},p} = \mathbf{I}_L \otimes \mathbf{h}_{\mathrm{T},p} \otimes \mathbf{h}_{\mathrm{S},p}$ is a matrix of size $MNL \times L$, and $\mathbf{I}_Q$ is an identity matrix of size $Q \times Q$. Then, the filter's output Z can be written as

$$\begin{aligned} Z = \underline{\mathbf{h}}_P^H \underline{\mathbf{y}} &= \sum_{p=1}^{P} (\mathbf{h}_{\mathrm{F},p} \otimes \mathbf{h}_{\mathrm{T},p} \otimes \mathbf{h}_{\mathrm{S},p})^H\, \underline{\mathbf{y}} \\ &= \sum_{p=1}^{P} \mathbf{h}_{\mathrm{S},p}^H \mathbf{H}_{\mathrm{TF},p}^H \underline{\mathbf{y}} = \underline{\mathbf{h}}_{\mathrm{S},P}^H \underline{\mathbf{y}}_{\mathrm{TF},P} \qquad (20) \\ &= \sum_{p=1}^{P} \mathbf{h}_{\mathrm{T},p}^H \mathbf{H}_{\mathrm{SF},p}^H \underline{\mathbf{y}} = \underline{\mathbf{h}}_{\mathrm{T},P}^H \underline{\mathbf{y}}_{\mathrm{SF},P} \qquad (21) \\ &= \sum_{p=1}^{P} \mathbf{h}_{\mathrm{F},p}^H \mathbf{H}_{\mathrm{ST},p}^H \underline{\mathbf{y}} = \underline{\mathbf{h}}_{\mathrm{F},P}^H \underline{\mathbf{y}}_{\mathrm{ST},P}, \qquad (22) \end{aligned}$$

where

$$\underline{\mathbf{h}}_{\mathrm{S},P} = \left[\ \mathbf{h}_{\mathrm{S},1}^T\ \ \mathbf{h}_{\mathrm{S},2}^T\ \ \cdots\ \ \mathbf{h}_{\mathrm{S},P}^T\ \right]^T, \tag{23}$$

$$\underline{\mathbf{h}}_{\mathrm{T},P} = \left[\ \mathbf{h}_{\mathrm{T},1}^T\ \ \mathbf{h}_{\mathrm{T},2}^T\ \ \cdots\ \ \mathbf{h}_{\mathrm{T},P}^T\ \right]^T, \tag{24}$$

$$\underline{\mathbf{h}}_{\mathrm{F},P} = \left[\ \mathbf{h}_{\mathrm{F},1}^T\ \ \mathbf{h}_{\mathrm{F},2}^T\ \ \cdots\ \ \mathbf{h}_{\mathrm{F},P}^T\ \right]^T, \tag{25}$$

$$\begin{aligned} \underline{\mathbf{y}}_{\mathrm{TF},P} &= \left[\ \underline{\mathbf{y}}^H \mathbf{H}_{\mathrm{TF},1}\ \ \underline{\mathbf{y}}^H \mathbf{H}_{\mathrm{TF},2}\ \ \cdots\ \ \underline{\mathbf{y}}^H \mathbf{H}_{\mathrm{TF},P}\ \right]^H \\ &= \underline{\mathbf{H}}_{\mathrm{TF},P} \underline{\mathbf{y}}, \end{aligned} \tag{26}$$

$$\begin{aligned} \underline{\mathbf{y}}_{\mathrm{SF},P} &= \left[\ \underline{\mathbf{y}}^H \mathbf{H}_{\mathrm{SF},1}\ \ \underline{\mathbf{y}}^H \mathbf{H}_{\mathrm{SF},2}\ \ \cdots\ \ \underline{\mathbf{y}}^H \mathbf{H}_{\mathrm{SF},P}\ \right]^H \\ &= \underline{\mathbf{H}}_{\mathrm{SF},P} \underline{\mathbf{y}}, \end{aligned} \tag{27}$$

$$\begin{aligned} \underline{\mathbf{y}}_{\mathrm{ST},P} &= \left[\ \underline{\mathbf{y}}^H \mathbf{H}_{\mathrm{ST},1}\ \ \underline{\mathbf{y}}^H \mathbf{H}_{\mathrm{ST},2}\ \ \cdots\ \ \underline{\mathbf{y}}^H \mathbf{H}_{\mathrm{ST},P}\ \right]^H \\ &= \underline{\mathbf{H}}_{\mathrm{ST},P} \underline{\mathbf{y}}, \end{aligned} \tag{28}$$

$$\underline{\mathbf{H}}_{\mathrm{TF},P} = \left[\ \mathbf{H}_{\mathrm{TF},1}\ \ \mathbf{H}_{\mathrm{TF},2}\ \ \cdots\ \ \mathbf{H}_{\mathrm{TF},P}\ \right]^H, \tag{29}$$

$$\underline{\mathbf{H}}_{\mathrm{SF},P} = \left[\ \mathbf{H}_{\mathrm{SF},1}\ \ \mathbf{H}_{\mathrm{SF},2}\ \ \cdots\ \ \mathbf{H}_{\mathrm{SF},P}\ \right]^H, \tag{30}$$

$$\underline{\mathbf{H}}_{\mathrm{ST},P} = \left[\ \mathbf{H}_{\mathrm{ST},1}\ \ \mathbf{H}_{\mathrm{ST},2}\ \ \cdots\ \ \mathbf{H}_{\mathrm{ST},P}\ \right]^H. \tag{31}$$

The sizes of $\underline{\mathbf{h}}_{\mathrm{S},P}$, $\underline{\mathbf{h}}_{\mathrm{T},P}$, $\underline{\mathbf{h}}_{\mathrm{F},P}$, $\underline{\mathbf{y}}_{\mathrm{TF},P}$, $\underline{\mathbf{y}}_{\mathrm{SF},P}$, $\underline{\mathbf{y}}_{\mathrm{ST},P}$, $\underline{\mathbf{H}}_{\mathrm{TF},P}$, $\underline{\mathbf{H}}_{\mathrm{SF},P}$ and $\underline{\mathbf{H}}_{\mathrm{ST},P}$ are, respectively, $MP \times 1$, $NP \times 1$, $LP \times 1$, $MP \times 1$, $NP \times 1$, $LP \times 1$, $MP \times MNL$, $NP \times MNL$ and $LP \times MNL$. Accordingly, instead of applying the filter $\underline{\mathbf{h}}$ of length MNL, we can utilize the sub-filters $\underline{\mathbf{h}}_{\mathrm{S},P}$ of length MP, $\underline{\mathbf{h}}_{\mathrm{T},P}$ of length NP and $\underline{\mathbf{h}}_{\mathrm{F},P}$ of length LP, which are much shorter than $\underline{\mathbf{h}}$ when P is small, to perform noise reduction.

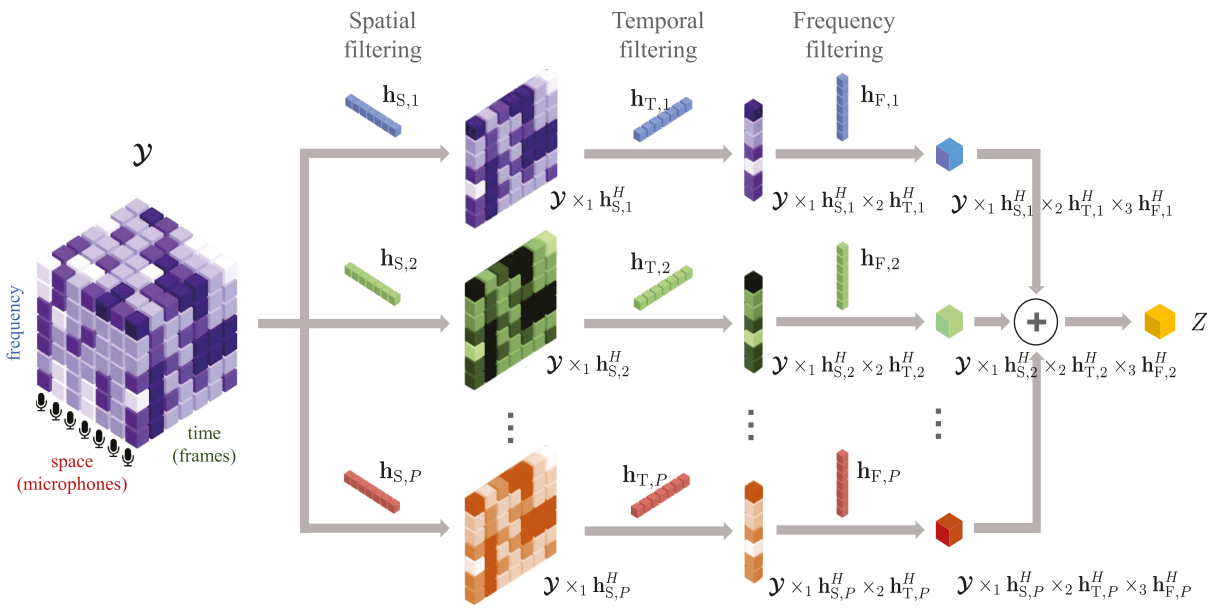


Fig. 1. The scheme of the filtering process with temporal, spatial and frequency sub-filters (filter decomposition). Note that the order of applying these sub-filters can be exchanged, leading to equivalent forms in, e.g., (20)-(22).

With some mathematical manipulations [36], [50], we can write Z as

$$Z = \underline{\mathbf{h}}_P^H \underline{\mathbf{y}} = \sum_{p=1}^{P} \left(\mathbf{h}_{\mathrm{F},p} \otimes \mathbf{h}_{\mathrm{T},p} \otimes \mathbf{h}_{\mathrm{S},p}\right)^H \underline{\mathbf{y}} = \sum_{p=1}^{P} \boldsymbol{\mathcal{Y}} \times_1 \mathbf{h}_{\mathrm{S},p}^H \times_2 \mathbf{h}_{\mathrm{T},p}^H \times_3 \mathbf{h}_{\mathrm{F},p}^H, \tag{32}$$

where $\boldsymbol{\mathcal{A}} \times_q \mathbf{B}$ means mode-q ($q = 1,\ 2,\ 3$) product of a tensor $\boldsymbol{\mathcal{A}}$ with a matrix $\mathbf{B}$. As can be seen from (32), the sub-filters $\mathbf{h}_{\mathrm{S},p},\ p = 1, 2, \ldots, P$, $\mathbf{h}_{\mathrm{T},p},\ p = 1, 2, \ldots, P$, and $\mathbf{h}_{\mathrm{F},p},\ p = 1, 2, \ldots, P$, process the multichannel noisy signal, respectively, along the spatial, temporal and frequency dimensions. As illustrated in Fig. 1, such a filtering process is significantly more parsimony compared to the filtering with the full filter $\underline{\mathbf{h}}$ when P is small. Note that among the different representations of the filtering operation, the result $\underline{\mathbf{h}}_P^H \underline{\mathbf{y}}$ is finally used for noise reduction.

It should be noted that we aim to recover X_1 rather than the entire signal vector $\underline{\mathbf{x}}$. The vector $\underline{\mathbf{x}}$ is thus decomposed into two parts [21], [29], namely, the correlated and uncorrelated (which will be treated as interference) components with respect to X_1, i.e.,

$$\underline{\mathbf{x}} = X_1 \underline{\boldsymbol{\rho}}_{\underline{\mathbf{x}} X_1} + \underline{\mathbf{x}}_{\mathrm{i}} = \underline{\mathbf{x}}_{\mathrm{d}} + \underline{\mathbf{x}}_{\mathrm{i}}, \tag{33}$$

where $\underline{\mathbf{x}}_{\mathrm{d}} = X_1 \underline{\boldsymbol{\rho}}_{\underline{\mathbf{x}} X_1}$ is the desired signal vector, $\underline{\mathbf{x}}_{\mathrm{i}} = \underline{\mathbf{x}} - \underline{\mathbf{x}}_{\mathrm{d}}$ is the interference vector,

$$\underline{\boldsymbol{\rho}}_{\underline{\mathbf{x}} X_1} = \frac{E\left(\underline{\mathbf{x}} X_1^*\right)}{E\left(|X_1|^2\right)} \tag{34}$$

is the partially normalized cross-correlation vector, of length MNL, between X_1 and $\underline{\mathbf{x}}$ (which is the first column of the correlation matrix of $\underline{\mathbf{x}}$ normalized by $E(|X_1|^2)$), and $(\cdot)^*$ denotes complex-conjugation. It is easy to verify that the interference signal vector, $\underline{\mathbf{x}}_{\mathrm{i}}$, and the desired signal, X_1, are uncorrelated with each other, i.e.,

$$E\left(\underline{\mathbf{x}}_{\mathrm{i}} X_1^*\right) = \mathbf{0}_{MNL}, \tag{35}$$

where $\mathbf{0}_{MNL}$ is an all-zero vector of length MNL.

Substituting (33) into (6), we get

$$\underline{\mathbf{y}} = X_1 \underline{\boldsymbol{\rho}}_{\underline{\mathbf{x}} X_1} + \underline{\mathbf{x}}_{\mathrm{i}} + \underline{\mathbf{v}} = \underline{\mathbf{x}}_{\mathrm{d}} + \underline{\mathbf{x}}_{\mathrm{i}} + \underline{\mathbf{v}}. \tag{36}$$

Based on (36), we can rewrite (20), (21) and (22) as

$$\begin{aligned}
Z &= \underline{\mathbf{h}}_P^H \underline{\mathbf{y}} \\
&= X_1 \underline{\mathbf{h}}_P^H \underline{\boldsymbol{\rho}}_{\underline{\mathbf{x}} X_1} + \underline{\mathbf{h}}_P^H \underline{\mathbf{x}}_{\mathrm{i}} + \underline{\mathbf{h}}_P^H \underline{\mathbf{v}} \\
&= \underline{\mathbf{h}}_{\mathrm{S},P}^H \underline{\mathbf{y}}_{\mathrm{TF},P} \\
&= X_1 \underline{\mathbf{h}}_{\mathrm{S},P}^H \underline{\boldsymbol{\rho}}_{\underline{\mathbf{x}} X_1,\mathrm{TF},P} + \underline{\mathbf{h}}_{\mathrm{S},P}^H \underline{\mathbf{x}}_{\mathrm{i},\mathrm{TF},P} + \underline{\mathbf{h}}_{\mathrm{S},P}^H \underline{\mathbf{v}}_{\mathrm{TF},P} \\
&= \underline{\mathbf{h}}_{\mathrm{T},P}^H \underline{\mathbf{y}}_{\mathrm{SF},P} \\
&= X_1 \underline{\mathbf{h}}_{\mathrm{T},P}^H \underline{\boldsymbol{\rho}}_{\underline{\mathbf{x}} X_1,\mathrm{SF},P} + \underline{\mathbf{h}}_{\mathrm{T},P}^H \underline{\mathbf{x}}_{\mathrm{i},\mathrm{SF},P} + \underline{\mathbf{h}}_{\mathrm{T},P}^H \underline{\mathbf{v}}_{\mathrm{SF},P} \\
&= \underline{\mathbf{h}}_{\mathrm{F},P}^H \underline{\mathbf{y}}_{\mathrm{ST},P} \\
&= X_1 \underline{\mathbf{h}}_{\mathrm{F},P}^H \underline{\boldsymbol{\rho}}_{\underline{\mathbf{x}} X_1,\mathrm{ST},P} + \underline{\mathbf{h}}_{\mathrm{F},P}^H \underline{\mathbf{x}}_{\mathrm{i},\mathrm{ST},P} + \underline{\mathbf{h}}_{\mathrm{F},P}^H \underline{\mathbf{v}}_{\mathrm{ST},P} \\
&= X_{\mathrm{fd}} + X_{\mathrm{ri}} + V_{\mathrm{rn}},
\end{aligned} \tag{37}$$

where $\underline{\boldsymbol{\rho}}_{\underline{\mathbf{x}} X_1,\mathrm{TF},P} = \underline{\mathbf{H}}_{\mathrm{TF},P} \underline{\boldsymbol{\rho}}_{\underline{\mathbf{x}} X_1}$, $\underline{\mathbf{x}}_{\mathrm{i},\mathrm{TF},P} = \underline{\mathbf{H}}_{\mathrm{TF},P} \underline{\mathbf{x}}_{\mathrm{i}}$, $\underline{\mathbf{v}}_{\mathrm{TF},P} = \underline{\mathbf{H}}_{\mathrm{TF},P} \underline{\mathbf{v}}$ are vectors of length MP, $\underline{\boldsymbol{\rho}}_{\underline{\mathbf{x}} X_1,\mathrm{SF},P} = \underline{\mathbf{H}}_{\mathrm{SF},P} \underline{\boldsymbol{\rho}}_{\underline{\mathbf{x}} X_1}$, $\underline{\mathbf{x}}_{\mathrm{i},\mathrm{SF},P} = \underline{\mathbf{H}}_{\mathrm{SF},P} \underline{\mathbf{x}}_{\mathrm{i}}$, $\underline{\mathbf{v}}_{\mathrm{SF},P} = \underline{\mathbf{H}}_{\mathrm{SF},P} \underline{\mathbf{v}}$ are vectors of length NP, $\underline{\boldsymbol{\rho}}_{\underline{\mathbf{x}} X_1,\mathrm{ST},P} = \underline{\mathbf{H}}_{\mathrm{ST},P} \underline{\boldsymbol{\rho}}_{\underline{\mathbf{x}} X_1}$,

$\underline{\mathbf{x}}_{\mathrm{i,ST},P} = \underline{\mathbf{H}}_{\mathrm{ST},P}\underline{\mathbf{x}}_{\mathrm{i}}$, $\underline{\mathbf{v}}_{\mathrm{ST},P} = \underline{\mathbf{H}}_{\mathrm{ST},P}\underline{\mathbf{v}}$ are vectors of length LP,

$$\begin{aligned} X_{\mathrm{fd}} &= X_1\underline{\mathbf{h}}_P^H\underline{\boldsymbol{\rho}}_{\underline{\mathbf{x}}X_1} = X_1\underline{\mathbf{h}}_{\mathrm{S},P}^H\underline{\boldsymbol{\rho}}_{\underline{\mathbf{x}}X_1,\mathrm{TF},P} \\ &= X_1\underline{\mathbf{h}}_{\mathrm{T},P}^H\underline{\boldsymbol{\rho}}_{\underline{\mathbf{x}}X_1,\mathrm{SF},P} = X_1\underline{\mathbf{h}}_{\mathrm{F},P}^H\underline{\boldsymbol{\rho}}_{\underline{\mathbf{x}}X_1,\mathrm{ST},P} \end{aligned} \tag{38}$$

is the filtered desired signal,

$$X_{\mathrm{ri}} = \underline{\mathbf{h}}_P^H\underline{\mathbf{x}}_{\mathrm{i}} = \underline{\mathbf{h}}_{\mathrm{S},P}^H\underline{\mathbf{x}}_{\mathrm{i,TF},P} = \underline{\mathbf{h}}_{\mathrm{T},P}^H\underline{\mathbf{x}}_{\mathrm{i,SF},P} = \underline{\mathbf{h}}_{\mathrm{F},P}^H\underline{\mathbf{x}}_{\mathrm{i,ST},P} \tag{39}$$

is the residual interference, and

$$V_{\mathrm{rn}} = \underline{\mathbf{h}}_P^H\underline{\mathbf{v}} = \underline{\mathbf{h}}_{\mathrm{S},P}^H\underline{\mathbf{v}}_{\mathrm{TF},P} = \underline{\mathbf{h}}_{\mathrm{T},P}^H\underline{\mathbf{v}}_{\mathrm{SF},P} = \underline{\mathbf{h}}_{\mathrm{F},P}^H\underline{\mathbf{v}}_{\mathrm{ST},P} \tag{40}$$

is the residual noise.

Obviously, the distortionless constraint(s) for noise reduction filter(s) in our context is

$$\begin{aligned} \underline{\mathbf{h}}_P^H\underline{\boldsymbol{\rho}}_{\underline{\mathbf{x}}X_1} &= \underline{\mathbf{h}}_{\mathrm{S},P}^H\underline{\boldsymbol{\rho}}_{\underline{\mathbf{x}}X_1,\mathrm{TF},P} = \underline{\mathbf{h}}_{\mathrm{T},P}^H\underline{\boldsymbol{\rho}}_{\underline{\mathbf{x}}X_1,\mathrm{SF},P} \\ &= \underline{\mathbf{h}}_{\mathrm{F},P}^H\underline{\boldsymbol{\rho}}_{\underline{\mathbf{x}}X_1,\mathrm{ST},P} = 1. \end{aligned} \tag{41}$$

It is clear that the three parts of the estimate of the desired signal are mutually uncorrelated. Hence, the variance of Z is given by

$$\phi_Z^2 = \underline{\mathbf{h}}_P^H\underline{\boldsymbol{\Phi}}_{\underline{\mathbf{y}}}\underline{\mathbf{h}}_P = \phi_{X_{\mathrm{fd}}}^2 + \phi_{X_{\mathrm{ri}}}^2 + \phi_{V_{\mathrm{rn}}}^2, \tag{42}$$

where

$$\phi_{X_{\mathrm{fd}}}^2 = \phi_{X_1}^2\left|\underline{\mathbf{h}}_P^H\underline{\boldsymbol{\rho}}_{\underline{\mathbf{x}}X_1}\right|^2, \tag{43}$$

$$\begin{aligned} \phi_{X_{\mathrm{ri}}}^2 &= \underline{\mathbf{h}}_P^H\underline{\boldsymbol{\Phi}}_{\underline{\mathbf{x}}_{\mathbf{i}}}\underline{\mathbf{h}}_P \\ &= \underline{\mathbf{h}}_P^H\underline{\boldsymbol{\Phi}}_{\underline{\mathbf{x}}}\underline{\mathbf{h}}_P - \phi_{X_1}^2\left|\underline{\mathbf{h}}_P^H\underline{\boldsymbol{\rho}}_{\underline{\mathbf{x}}X_1}\right|^2, \end{aligned} \tag{44}$$

$$\phi_{V_{\mathrm{rn}}}^2 = \underline{\mathbf{h}}_P^H\underline{\boldsymbol{\Phi}}_{\underline{\mathbf{v}}}\underline{\mathbf{h}}_P, \tag{45}$$

$\phi_{X_1}^2 = E(|X_1|^2)$, $\underline{\boldsymbol{\Phi}}_{\underline{\boldsymbol{\alpha}}} = E(\underline{\boldsymbol{\alpha}}\,\underline{\boldsymbol{\alpha}}^H)$ is the correlation matrix of $\underline{\boldsymbol{\alpha}} \in \{\underline{\mathbf{y}}, \underline{\mathbf{x}}, \underline{\mathbf{x}}_{\mathrm{i}}, \underline{\mathbf{v}}\}$ of size $MNL \times MNL$, and $\underline{\boldsymbol{\Phi}}_{\underline{\mathbf{x}}_{\mathbf{i}}} = \underline{\boldsymbol{\Phi}}_{\underline{\mathbf{x}}} - \phi_{X_1}^2\underline{\boldsymbol{\rho}}_{\underline{\mathbf{x}}X_1}\underline{\boldsymbol{\rho}}_{\underline{\mathbf{x}}X_1}^H$.

III. Performance Measures

Since X_1 is considered as the desired signal, the input signal-to-noise ratio (iSNR) is defined as

$$\mathrm{iSNR} = \frac{\phi_{X_1}^2}{\phi_{V_1}^2}, \tag{46}$$

where $\phi_{V_1}^2 = E(|V_1|^2)$ is the variance of the noise at the reference microphone. The output signal-to-noise ratio (oSNR) is defined as

$$\mathrm{oSNR}\left(\underline{\mathbf{h}}_P\right) = \frac{\phi_{X_{\mathrm{fd}}}^2}{\phi_{X_{\mathrm{ri}}}^2 + \phi_{V_{\mathrm{rn}}}^2} = \frac{\phi_{X_1}^2\left(\underline{\mathbf{h}}_P^H\underline{\boldsymbol{\rho}}_{\underline{\mathbf{x}}X_1}\right)^2}{\underline{\mathbf{h}}_P^H\underline{\boldsymbol{\Phi}}_{\mathrm{in}}\underline{\mathbf{h}}_P}, \tag{47}$$

where $\underline{\boldsymbol{\Phi}}_{\mathrm{in}} = \underline{\boldsymbol{\Phi}}_{\underline{\mathbf{x}}_{\mathbf{i}}} + \underline{\boldsymbol{\Phi}}_{\underline{\mathbf{v}}}$ is the correlation matrix of the interference-plus-noise.

We define the speech distortion index as

$$\begin{aligned} \upsilon_{\mathrm{sd}}(\underline{\mathbf{h}}_P) &= \frac{E\left(|X_{\mathrm{fd}} - X_1|^2\right)}{E\left(|X_1|^2\right)} = \left|\underline{\mathbf{h}}_P^H\underline{\boldsymbol{\rho}}_{\underline{\mathbf{x}}X_1} - 1\right|^2 \\ &= \left|\underline{\mathbf{h}}_{\mathrm{S},P}^H\underline{\boldsymbol{\rho}}_{\underline{\mathbf{x}}X_1,\mathrm{TF},P} - 1\right|^2 = \left|\underline{\mathbf{h}}_{\mathrm{T},P}^H\underline{\boldsymbol{\rho}}_{\underline{\mathbf{x}}X_1,\mathrm{SF},P} - 1\right|^2 \\ &= \left|\underline{\mathbf{h}}_{\mathrm{F},P}^H\underline{\boldsymbol{\rho}}_{\underline{\mathbf{x}}X_1,\mathrm{ST},P} - 1\right|^2. \end{aligned} \tag{48}$$

When the distortionless constraint(s) is satisfied, the speech distortion index is equal to zero.

The mean square error (MSE) between the desired and estimated speech signals is defined as

$$\begin{aligned} J(\underline{\mathbf{h}}_P) &= E\left(\left|X_1 - \underline{\mathbf{h}}_P^H\underline{\mathbf{y}}\right|^2\right) \\ &= \phi_{X_1}^2 - 2\phi_{X_1}^2\underline{\mathbf{h}}_P^H\underline{\boldsymbol{\rho}}_{\underline{\mathbf{x}}X_1} + \underline{\mathbf{h}}_P^H\underline{\boldsymbol{\Phi}}_{\underline{\mathbf{y}}}\underline{\mathbf{h}}_P \\ &= J_{\mathrm{d}}(\underline{\mathbf{h}}_P) + J_{\mathrm{r}}(\underline{\mathbf{h}}_P), \end{aligned} \tag{49}$$

where

$$J_{\mathrm{d}}(\underline{\mathbf{h}}_P) = E\left(\left|X_1 - \underline{\mathbf{h}}_P^H\underline{\mathbf{x}}_{\mathrm{d}}\right|^2\right) = \phi_{X_1}^2\upsilon_{\mathrm{sd}}(\underline{\mathbf{h}}_P) \tag{50}$$

and

$$\begin{aligned} J_{\mathrm{r}}(\underline{\mathbf{h}}_P) &= E\left(\left|\underline{\mathbf{h}}_P^H\underline{\mathbf{x}}_{\mathrm{i}}\right|^2\right) + E\left(\left|\underline{\mathbf{h}}_P^H\underline{\mathbf{v}}\right|^2\right) \\ &= \underline{\mathbf{h}}_P^H\underline{\boldsymbol{\Phi}}_{\mathrm{in}}\underline{\mathbf{h}}_P = \phi_{X_{\mathrm{ri}}}^2 + \phi_{V_{\mathrm{rn}}}^2. \end{aligned} \tag{51}$$

IV. Iterative Noise Reduction Filters Based on the Kronecker Product Decomposition

In order to take advantage of the structure in (37) for noise reduction, we rewrite the MSE as the following equivalent forms:

$$J(\underline{\mathbf{h}}_P)$$

$$= \phi_{X_1}^2 - 2\phi_{X_1}^2\underline{\mathbf{h}}_{\mathrm{S},P}^H\underline{\boldsymbol{\rho}}_{\underline{\mathbf{x}}X_1,\mathrm{TF},P} + \underline{\mathbf{h}}_{\mathrm{S},P}^H\underline{\boldsymbol{\Phi}}_{\underline{\mathbf{y}}_{\mathrm{TF},P}}\underline{\mathbf{h}}_{\mathrm{S},P} \tag{52}$$

$$= \phi_{X_1}^2 - 2\phi_{X_1}^2\underline{\mathbf{h}}_{\mathrm{T},P}^H\underline{\boldsymbol{\rho}}_{\underline{\mathbf{x}}X_1,\mathrm{SF},P} + \underline{\mathbf{h}}_{\mathrm{T},P}^H\underline{\boldsymbol{\Phi}}_{\underline{\mathbf{y}}_{\mathrm{SF},P}}\underline{\mathbf{h}}_{\mathrm{T},P} \tag{53}$$

$$= \phi_{X_1}^2 - 2\phi_{X_1}^2\underline{\mathbf{h}}_{\mathrm{F},P}^H\underline{\boldsymbol{\rho}}_{\underline{\mathbf{x}}X_1,\mathrm{ST},P} + \underline{\mathbf{h}}_{\mathrm{F},P}^H\underline{\boldsymbol{\Phi}}_{\underline{\mathbf{y}}_{\mathrm{ST},P}}\underline{\mathbf{h}}_{\mathrm{F},P}, \tag{54}$$

where

$$\underline{\boldsymbol{\Phi}}_{\underline{\mathbf{y}}_{\mathrm{TF},P}} = \underline{\mathbf{H}}_{\mathrm{TF},P}\underline{\boldsymbol{\Phi}}_{\underline{\mathbf{y}}}\underline{\mathbf{H}}_{\mathrm{TF},P}^H, \tag{55}$$

$$\underline{\boldsymbol{\rho}}_{\underline{\mathbf{x}}X_1,\mathrm{TF},P} = \underline{\mathbf{H}}_{\mathrm{TF},P}\underline{\boldsymbol{\rho}}_{\underline{\mathbf{x}}X_1}, \tag{56}$$

$$\underline{\boldsymbol{\Phi}}_{\underline{\mathbf{y}}_{\mathrm{SF},P}} = \underline{\mathbf{H}}_{\mathrm{SF},P}\underline{\boldsymbol{\Phi}}_{\underline{\mathbf{y}}}\underline{\mathbf{H}}_{\mathrm{SF},P}^H, \tag{57}$$

$$\underline{\boldsymbol{\rho}}_{\underline{\mathbf{x}}X_1,\mathrm{SF},P} = \underline{\mathbf{H}}_{\mathrm{SF},P}\underline{\boldsymbol{\rho}}_{\underline{\mathbf{x}}X_1}, \tag{58}$$

$$\underline{\boldsymbol{\Phi}}_{\underline{\mathbf{y}}_{\mathrm{ST},P}} = \underline{\mathbf{H}}_{\mathrm{ST},P}\underline{\boldsymbol{\Phi}}_{\underline{\mathbf{y}}}\underline{\mathbf{H}}_{\mathrm{ST},P}^H, \tag{59}$$

$$\underline{\boldsymbol{\rho}}_{\underline{\mathbf{x}}X_1,\mathrm{ST},P} = \underline{\mathbf{H}}_{\mathrm{ST},P}\underline{\boldsymbol{\rho}}_{\underline{\mathbf{x}}X_1}. \tag{60}$$

It should be noticed that the sizes of the matrices $\underline{\boldsymbol{\Phi}}_{\underline{\mathbf{y}}_{\mathrm{TF},P}}$ ($MP \times MP$), $\underline{\boldsymbol{\Phi}}_{\underline{\mathbf{y}}_{\mathrm{SF},P}}$ ($NP \times NP$) and $\underline{\boldsymbol{\Phi}}_{\underline{\mathbf{y}}_{\mathrm{ST},P}}$ ($LP \times LP$) are much smaller than that of the matrix $\underline{\boldsymbol{\Phi}}_{\underline{\mathbf{y}}}$ ($MNL \times MNL$) when P is small. Consequently, the computational complexity

involved in the matrix inversion of $\underline{\mathbf{\Phi}}_{\underline{\mathbf{y}}_{\mathrm{TF},P}}$, $\underline{\mathbf{\Phi}}_{\underline{\mathbf{y}}_{\mathrm{SF},P}}$ and $\underline{\mathbf{\Phi}}_{\underline{\mathbf{y}}_{\mathrm{ST},P}}$ is much lower than that of $\underline{\mathbf{\Phi}}_{\underline{\mathbf{y}}}$. Moreover, much less observations are needed to achieve confident estimates of $\underline{\mathbf{\Phi}}_{\underline{\mathbf{y}}_{\mathrm{TF},P}}$, $\underline{\mathbf{\Phi}}_{\underline{\mathbf{y}}_{\mathrm{SF},P}}$ and $\underline{\mathbf{\Phi}}_{\underline{\mathbf{y}}_{\mathrm{ST},P}}$ than $\underline{\mathbf{\Phi}}_{\underline{\mathbf{y}}}$. As a result, the noise reduction algorithms developed based on the matrices $\underline{\mathbf{\Phi}}_{\underline{\mathbf{y}}_{\mathrm{TF},P}}$, $\underline{\mathbf{\Phi}}_{\underline{\mathbf{y}}_{\mathrm{SF},P}}$ and $\underline{\mathbf{\Phi}}_{\underline{\mathbf{y}}_{\mathrm{ST},P}}$ have significantly lower computational complexity and are more efficient in tracking the temporal/spatial nonstationarity of the speech and noise signals than traditional noise reduction algorithms based on the matrix $\underline{\mathbf{\Phi}}_{\underline{\mathbf{y}}}$ [21], [29].

With the forms of the MSEs in (52), (53) and (54), it is difficult to deduce closed-form solutions of $\underline{\mathbf{h}}_{\mathrm{S},P}$, $\underline{\mathbf{h}}_{\mathrm{T},P}$ and $\underline{\mathbf{h}}_{\mathrm{F},P}$. We propose to develop iterative algorithms for multichannel noise reduction based on the general optimization routine of the Kronecker filter. The convergence of the general routine has been studied in [45], [51], [52].

A. Iterative Wiener Filter

In order to derive the iterative Wiener filter, we write (52), (53) and (54) under the assumption that two of the filters $\underline{\mathbf{h}}_{\mathrm{S},P}$, $\underline{\mathbf{h}}_{\mathrm{T},P}$ and $\underline{\mathbf{h}}_{\mathrm{F},P}$ are fixed, i.e.,

$$J(\underline{\mathbf{h}}_{\mathrm{S},P}|\underline{\mathbf{h}}_{\mathrm{T},P},\underline{\mathbf{h}}_{\mathrm{F},P}) = \phi_{X_1}^2 - 2\phi_{X_1}^2\underline{\mathbf{h}}_{\mathrm{S},P}^H\underline{\boldsymbol{\rho}}_{\underline{\mathbf{x}}X_1,\mathrm{TF},P} + \underline{\mathbf{h}}_{\mathrm{S},P}^H\underline{\mathbf{\Phi}}_{\underline{\mathbf{y}}_{\mathrm{TF},P}}\underline{\mathbf{h}}_{\mathrm{S},P}, \tag{61}$$

$$J(\underline{\mathbf{h}}_{\mathrm{T},P}|\underline{\mathbf{h}}_{\mathrm{S},P},\underline{\mathbf{h}}_{\mathrm{F},P}) = \phi_{X_1}^2 - 2\phi_{X_1}^2\underline{\mathbf{h}}_{\mathrm{T},P}^H\underline{\boldsymbol{\rho}}_{\underline{\mathbf{x}}X_1,\mathrm{SF},P} + \underline{\mathbf{h}}_{\mathrm{T},P}^H\underline{\mathbf{\Phi}}_{\underline{\mathbf{y}}_{\mathrm{SF},P}}\underline{\mathbf{h}}_{\mathrm{T},P}, \tag{62}$$

$$J(\underline{\mathbf{h}}_{\mathrm{F},P}|\underline{\mathbf{h}}_{\mathrm{S},P},\underline{\mathbf{h}}_{\mathrm{T},P}) = \phi_{X_1}^2 - 2\phi_{X_1}^2\underline{\mathbf{h}}_{\mathrm{F},P}^H\underline{\boldsymbol{\rho}}_{\underline{\mathbf{x}}X_1,\mathrm{ST},P} + \underline{\mathbf{h}}_{\mathrm{F},P}^H\underline{\mathbf{\Phi}}_{\underline{\mathbf{y}}_{\mathrm{ST},P}}\underline{\mathbf{h}}_{\mathrm{F},P}. \tag{63}$$

Consider the initialization of $\underline{\mathbf{h}}_{\mathrm{T},P}^{(0)}$ as

$$\underline{\mathbf{h}}_{\mathrm{T},P}^{(0)} = \left[\ \mathbf{h}_{\mathrm{TW},1}^T\ \ \mathbf{h}_{\mathrm{TW},2}^T\ \ \cdots\ \ \mathbf{h}_{\mathrm{TW},P}^T\ \right]^T, \tag{64}$$

where

$$\mathbf{h}_{\mathrm{TW},p} = \frac{\phi_{X_p}^2\mathbf{\Phi}_{\mathrm{in,T},p}^{-1}\boldsymbol{\rho}_{\mathbf{x}_{\mathrm{T},p}X_p}}{1+\phi_{X_p}^2\boldsymbol{\rho}_{\mathbf{x}_{\mathrm{T},p}X_p}^H\mathbf{\Phi}_{\mathrm{in,T},p}^{-1}\boldsymbol{\rho}_{\mathbf{x}_{\mathrm{T},p}X_p}}, \quad p=1,2,\ldots,P, \tag{65}$$

$\mathbf{\Phi}_{\mathrm{in,T},p} = \mathbf{\Phi}_{\mathbf{y}_{\mathrm{T},p}} - \phi_{X_p}^2\boldsymbol{\rho}_{\mathbf{x}_{\mathrm{T},p}X_p}\boldsymbol{\rho}_{\mathbf{x}_{\mathrm{T},p}X_p}^H$, $\mathbf{\Phi}_{\mathbf{y}_{\mathrm{T},p}} = E(\mathbf{y}_{\mathrm{T},p}\mathbf{y}_{\mathrm{T},p}^H)$, $\boldsymbol{\rho}_{\mathbf{x}_{\mathrm{T},p}X_p} = \dfrac{E(\mathbf{x}_{\mathrm{T},p}X_p^*)}{E(|X_p|^2)}$,

$$\mathbf{y}_{\mathrm{T},p} = \boldsymbol{\mathcal{Y}}_{p:1} = \left[\ Y_p(k,n)\ \ Y_p(k,n-1)\ \ \cdots\ \ Y_p(k,n-N+1)\right]^T, \tag{66}$$

$$\mathbf{x}_{\mathrm{T},p} = \boldsymbol{\mathcal{X}}_{p:1} = \left[\ X_p(k,n)\ \ X_p(k,n-1)\ \ \cdots\ \ X_p(k,n-N+1)\right]^T, \tag{67}$$

and $\phi_{X_p}^2 = E(|X_p|^2)$ with X_p denoting $X_p(k,n)$. Consider the initialization of $\underline{\mathbf{h}}_{\mathrm{F},P}^{(0)}$ as

$$\underline{\mathbf{h}}_{\mathrm{F},P}^{(0)} = \left[\ \mathbf{h}_{\mathrm{FW},1}^T\ \ \mathbf{h}_{\mathrm{FW},2}^T\ \ \cdots\ \ \mathbf{h}_{\mathrm{FW},P}^T\ \right]^T, \tag{68}$$

where

$$\mathbf{h}_{\mathrm{FW},p} = \frac{\phi_{X_p}^2\mathbf{\Phi}_{\mathrm{in,F},p}^{-1}\boldsymbol{\rho}_{\mathbf{x}_{\mathrm{F},p}X_p}}{1+\phi_{X_p}^2\boldsymbol{\rho}_{\mathbf{x}_{\mathrm{F},p}X_p}^H\mathbf{\Phi}_{\mathrm{in,F},p}^{-1}\boldsymbol{\rho}_{\mathbf{x}_{\mathrm{F},p}X_p}}, \quad p=1,2,\ldots,P, \tag{69}$$

$\mathbf{\Phi}_{\mathrm{in,F},p} = \mathbf{\Phi}_{\mathbf{y}_{\mathrm{F},p}} - \phi_{X_p}^2\boldsymbol{\rho}_{\mathbf{x}_{\mathrm{F},p}X_p}\boldsymbol{\rho}_{\mathbf{x}_{\mathrm{F},p}X_p}^H$, $\mathbf{\Phi}_{\mathbf{y}_{\mathrm{F},p}} = E(\mathbf{y}_{\mathrm{F},p}\mathbf{y}_{\mathrm{F},p}^H)$, $\boldsymbol{\rho}_{\mathbf{x}_{\mathrm{F},p}X_p} = \dfrac{E(\mathbf{x}_{\mathrm{F},p}X_p^*)}{E(|X_p|^2)}$,

$$\mathbf{y}_{\mathrm{F},p} = \boldsymbol{\mathcal{Y}}_{p1:} = \left[\ Y_p(k-K_k^-,n)\ \ \cdots\ \ Y_p(k-1,n)\ \ Y_p(k,n)\ \ Y_p(k+1,n)\ \ \cdots\ \ Y_p(k+K_k^+,n)\ \right]^T, \tag{70}$$

and

$$\mathbf{x}_{\mathrm{F},p} = \boldsymbol{\mathcal{X}}_{p1:} = \left[\ X_p(k-K_k^-,n)\ \ \cdots\ \ X_p(k-1,n)\ \ X_p(k,n)\ \ X_p(k+1,n)\ \ \cdots\ \ X_p(k+K_k^+,n)\ \right]^T. \tag{71}$$

Note that $\mathbf{h}_{\mathrm{TW},p}$, $p=1,\ 2,\ \ldots,\ P$, is the Wiener filter of length N of the pth channel when only the inter-frame information is taken into account, and $\mathbf{h}_{\mathrm{FW},p}$, $p=1,\ 2,\ \ldots,\ P$, is the Wiener filter of length L of the pth channel when only the inter-band information is taken into account [29]. Actually, it is a simple and straightforward way to initialize $\underline{\mathbf{h}}_{\mathrm{T},P}^{(0)}$ and $\underline{\mathbf{h}}_{\mathrm{F},P}^{(0)}$ using only the signal of the first microphone, but extensive simulations show that the above initialization method converges faster.

Then, iteratively solving the cost functions (61), (62) and (63) yields the iterative Wiener sub-filters

$$\underline{\mathbf{h}}_{\mathrm{S},P}^{(n)} = \frac{\phi_{X_1}^2\left(\mathbf{\Phi}_{\mathrm{in,TF},P}^{(n-1)}\right)^{-1}\underline{\boldsymbol{\rho}}_{\underline{\mathbf{x}}X_1,\mathrm{TF},P}^{(n-1)}}{1+\phi_{X_1}^2\left(\underline{\boldsymbol{\rho}}_{\underline{\mathbf{x}}X_1,\mathrm{TF},P}^{(n-1)}\right)^H\left(\mathbf{\Phi}_{\mathrm{in,TF},P}^{(n-1)}\right)^{-1}\underline{\boldsymbol{\rho}}_{\underline{\mathbf{x}}X_1,\mathrm{TF},P}^{(n-1)}}, \tag{72}$$

$$\underline{\mathbf{h}}_{\mathrm{T},P}^{(n)} = \frac{\phi_{X_1}^2\left(\mathbf{\Phi}_{\mathrm{in,SF},P}^{(n)}\right)^{-1}\underline{\boldsymbol{\rho}}_{\underline{\mathbf{x}}X_1,\mathrm{SF},P}^{(n)}}{1+\phi_{X_1}^2\left(\underline{\boldsymbol{\rho}}_{\underline{\mathbf{x}}X_1,\mathrm{SF},P}^{(n)}\right)^H\left(\mathbf{\Phi}_{\mathrm{in,SF},P}^{(n)}\right)^{-1}\underline{\boldsymbol{\rho}}_{\underline{\mathbf{x}}X_1,\mathrm{SF},P}^{(n)}} \tag{73}$$

and

$$\underline{\mathbf{h}}_{\mathrm{F},P}^{(n)} = \frac{\phi_{X_1}^2\left(\mathbf{\Phi}_{\mathrm{in,ST},P}^{(n)}\right)^{-1}\underline{\boldsymbol{\rho}}_{\underline{\mathbf{x}}X_1,\mathrm{ST},P}^{(n)}}{1+\phi_{X_1}^2\left(\underline{\boldsymbol{\rho}}_{\underline{\mathbf{x}}X_1,\mathrm{ST},P}^{(n)}\right)^H\left(\mathbf{\Phi}_{\mathrm{in,ST},P}^{(n)}\right)^{-1}\underline{\boldsymbol{\rho}}_{\underline{\mathbf{x}}X_1,\mathrm{ST},P}^{(n)}}, \tag{74}$$

where $\mathbf{\Phi}_{\mathrm{in,TF},P}^{(n-1)}$, $\mathbf{\Phi}_{\mathrm{in,SF},P}^{(n)}$, $\mathbf{\Phi}_{\mathrm{in,ST},P}^{(n)}$, $\underline{\boldsymbol{\rho}}_{\underline{\mathbf{x}}X_1,\mathrm{TF},P}^{(n-1)}$, $\underline{\boldsymbol{\rho}}_{\underline{\mathbf{x}}X_1,\mathrm{SF},P}^{(n)}$ and $\underline{\boldsymbol{\rho}}_{\underline{\mathbf{x}}X_1,\mathrm{ST},P}^{(n)}$ are defined, respectively, in (124), (125), (126), (115), (117) and (119), and the detailed derivation steps are provided in Appendix A.

Finally, the iterative Wiener filter at iteration n is constructed as

$$\underline{\mathbf{h}}_{\mathrm{W},P} = \sum_{p=1}^{P}\mathbf{h}_{\mathrm{F},p}^{(n)}\otimes\mathbf{h}_{\mathrm{T},p}^{(n)}\otimes\mathbf{h}_{\mathrm{S},p}^{(n)}. \tag{75}$$

A summary of the algorithm for implementing the iterative Wiener filter is provided in Algorithm 1 as an example, where

O is the iteration number, and the algorithms for implementing the iterative MVDR and tradeoff filters can be immediately summarized in a similar manner.

Algorithm 1: Algorithm for Implementing the Iterative Wiener filter

Input: The estimated version of $\mathbf{\Phi}_{\underline{\mathbf{x}}}$ and $\mathbf{\Phi}_{\underline{\mathbf{v}}}$;
Output: The iterative Wiener filter $\underline{\mathbf{h}}_{\mathrm{W},P}$;

```
Compute ρ_{x X_1} by equation (34);
Compute Φ_{x_i} by Φ_{x_i} = Φ_x − φ²_{X_1} ρ_{x X_1} ρ^H_{x X_1}, and compute Φ_in by Φ_in = Φ_{x_i} + Φ_v;
Initialize h^(0)_{T,P} and h^(0)_{F,P} by equations (64) and (68), respectively;
for n = 1 : O do
    Use h^(n−1)_{T,P} and h^(n−1)_{F,P} to construct H^(n−1)_{TF,P} by equation (29);
    Compute Φ^(n−1)_{y_{TF,P}} by equation (55);
    Compute ρ^(n−1)_{x X_1,TF,P} by equation (56);
    Compute h^(n)_{S,P} by equation (72);
    Use h^(n)_{S,P} and h^(n−1)_{F,P} to construct H^(n)_{SF,P} by equation (30);
    Compute Φ^(n)_{y_{SF,P}} by equation (57);
    Compute ρ^(n)_{x X_1,SF,P} by equation (58);
    Compute h^(n)_{T,P} by equation (73);
    Use h^(n)_{S,P} and h^(n)_{T,P} to construct H^(n)_{ST,P} by equation (31);
    Compute Φ^(n)_{y_{ST,P}} by equation (59);
    Compute ρ^(n)_{x X_1,ST,P} by equation (60);
    Compute h^(n)_{F,P} by equation (74);
end for
Construct h_{W,P} by equation (75);
```

B. Iterative MVDR Filter

The iterative MVDR filter can be deduced by minimizing the variances of the residual interference-plus-noise under the distortionless constraints. Mathematically, this is equivalent to

$$\min_{\underline{\mathbf{h}}_{\mathrm{S},P}^{(n)}} \left(\underline{\mathbf{h}}_{\mathrm{S},P}^{(n)}\right)^H \mathbf{\Phi}_{\mathrm{in,TF},P}^{(n-1)} \underline{\mathbf{h}}_{\mathrm{S},P}^{(n)} \quad \text{s. t.} \quad \left(\underline{\mathbf{h}}_{\mathrm{S},P}^{(n)}\right)^H \underline{\boldsymbol{\rho}}_{\underline{\mathbf{x}}X_1,\mathrm{TF},P}^{(n-1)} = 1, \tag{76}$$

$$\min_{\underline{\mathbf{h}}_{\mathrm{T},P}^{(n)}} \left(\underline{\mathbf{h}}_{\mathrm{T},P}^{(n)}\right)^H \mathbf{\Phi}_{\mathrm{in,SF},P}^{(n)} \underline{\mathbf{h}}_{\mathrm{T},P}^{(n)} \quad \text{s. t.} \quad \left(\underline{\mathbf{h}}_{\mathrm{T},P}^{(n)}\right)^H \underline{\boldsymbol{\rho}}_{\underline{\mathbf{x}}X_1,\mathrm{SF},P}^{(n)} = 1, \tag{77}$$

$$\min_{\underline{\mathbf{h}}_{\mathrm{F},P}^{(n)}} \left(\underline{\mathbf{h}}_{\mathrm{F},P}^{(n)}\right)^H \mathbf{\Phi}_{\mathrm{in,ST},P}^{(n)} \underline{\mathbf{h}}_{\mathrm{F},P}^{(n)} \quad \text{s. t.} \quad \left(\underline{\mathbf{h}}_{\mathrm{F},P}^{(n)}\right)^H \underline{\boldsymbol{\rho}}_{\underline{\mathbf{x}}X_1,\mathrm{ST},P}^{(n)} = 1. \tag{78}$$

Consider the initialization of $\underline{\mathbf{h}}_{\mathrm{T},P}^{(0)}$ as

$$\underline{\mathbf{h}}_{\mathrm{T},P}^{(0)} = \left[\, \mathbf{h}_{\mathrm{TMVDR},1}^T \;\; \mathbf{h}_{\mathrm{TMVDR},2}^T \;\; \cdots \;\; \mathbf{h}_{\mathrm{TMVDR},P}^T \,\right]^T, \tag{79}$$

where

$$\mathbf{h}_{\mathrm{TMVDR},p} = \frac{\mathbf{\Phi}_{\mathrm{in,T},p}^{-1} \boldsymbol{\rho}_{\mathbf{x}_{\mathrm{T},p}X_p}}{\boldsymbol{\rho}_{\mathbf{x}_{\mathrm{T},p}X_p}^H \mathbf{\Phi}_{\mathrm{in,T},p}^{-1} \boldsymbol{\rho}_{\mathbf{x}_{\mathrm{T},p}X_p}}, \quad p = 1, 2, \ldots, P, \tag{80}$$

and the initialization of $\underline{\mathbf{h}}_{\mathrm{F},P}^{(0)}$ as

$$\underline{\mathbf{h}}_{\mathrm{F},P}^{(0)} = \left[\, \mathbf{h}_{\mathrm{FMVDR},1}^T \;\; \mathbf{h}_{\mathrm{FMVDR},2}^T \;\; \cdots \;\; \mathbf{h}_{\mathrm{FMVDR},P}^T \,\right]^T, \tag{81}$$

where

$$\mathbf{h}_{\mathrm{FMVDR},p} = \frac{\mathbf{\Phi}_{\mathrm{in,F},p}^{-1} \boldsymbol{\rho}_{\mathbf{x}_{\mathrm{F},p}X_p}}{\boldsymbol{\rho}_{\mathbf{x}_{\mathrm{F},p}X_p}^H \mathbf{\Phi}_{\mathrm{in,F},p}^{-1} \boldsymbol{\rho}_{\mathbf{x}_{\mathrm{F},p}X_p}}, \quad p = 1, 2, \ldots, P. \tag{82}$$

Note that $\mathbf{h}_{\mathrm{TMVDR},p},\ p = 1,\ 2,\ \ldots,\ P$, is the MVDR filter of length N of the pth channel when only the inter-frame information is taken into account, and $\mathbf{h}_{\mathrm{FMVDR},p},\ p = 1,\ 2,\ \ldots,\ P$, is the MVDR filter of length L of the pth channel when only the inter-band information is taken into account [29].

Then, solving the cost functions (76), (77) and (78) in an iterative way gives the following iterative MVDR sub-filters:

$$\underline{\mathbf{h}}_{\mathrm{S},P}^{(n)} = \frac{\left(\mathbf{\Phi}_{\mathrm{in,TF},P}^{(n-1)}\right)^{-1} \underline{\boldsymbol{\rho}}_{\underline{\mathbf{x}}X_1,\mathrm{TF},P}^{(n-1)}}{\left(\underline{\boldsymbol{\rho}}_{\underline{\mathbf{x}}X_1,\mathrm{TF},P}^{(n-1)}\right)^H \left(\mathbf{\Phi}_{\mathrm{in,TF},P}^{(n-1)}\right)^{-1} \underline{\boldsymbol{\rho}}_{\underline{\mathbf{x}}X_1,\mathrm{TF},P}^{(n-1)}}, \tag{83}$$

$$\underline{\mathbf{h}}_{\mathrm{T},P}^{(n)} = \frac{\left(\mathbf{\Phi}_{\mathrm{in,SF},P}^{(n)}\right)^{-1} \underline{\boldsymbol{\rho}}_{\underline{\mathbf{x}}X_1,\mathrm{SF},P}^{(n)}}{\left(\underline{\boldsymbol{\rho}}_{\underline{\mathbf{x}}X_1,\mathrm{SF},P}^{(n)}\right)^H \left(\mathbf{\Phi}_{\mathrm{in,SF},P}^{(n)}\right)^{-1} \underline{\boldsymbol{\rho}}_{\underline{\mathbf{x}}X_1,\mathrm{SF},P}^{(n)}} \tag{84}$$

and

$$\underline{\mathbf{h}}_{\mathrm{F},P}^{(n)} = \frac{\left(\mathbf{\Phi}_{\mathrm{in,ST},P}^{(n)}\right)^{-1} \underline{\boldsymbol{\rho}}_{\underline{\mathbf{x}}X_1,\mathrm{ST},P}^{(n)}}{\left(\underline{\boldsymbol{\rho}}_{\underline{\mathbf{x}}X_1,\mathrm{ST},P}^{(n)}\right)^H \left(\mathbf{\Phi}_{\mathrm{in,ST},P}^{(n)}\right)^{-1} \underline{\boldsymbol{\rho}}_{\underline{\mathbf{x}}X_1,\mathrm{ST},P}^{(n)}}. \tag{85}$$

Finally, the iterative MVDR filter at the nth iteration is given by

$$\underline{\mathbf{h}}_{\mathrm{MVDR},P} = \sum_{p=1}^{P} \mathbf{h}_{\mathrm{F},p}^{(n)} \otimes \mathbf{h}_{\mathrm{T},p}^{(n)} \otimes \mathbf{h}_{\mathrm{S},p}^{(n)}. \tag{86}$$

Another equivalent form of the iterative MVDR filter can be obtained by replacing $\mathbf{\Phi}_{\mathrm{in,TF},P}^{(n-1)}$, $\mathbf{\Phi}_{\mathrm{in,SF},P}^{(n)}$ and $\mathbf{\Phi}_{\mathrm{in,ST},P}^{(n)}$ in the cost functions by $\mathbf{\Phi}_{\underline{\mathbf{y}}_{\mathrm{TF},P}}^{(n-1)}$, $\mathbf{\Phi}_{\underline{\mathbf{y}}_{\mathrm{SF},P}}^{(n)}$ and $\mathbf{\Phi}_{\underline{\mathbf{y}}_{\mathrm{ST},P}}^{(n)}$, respectively, which are defined in (121), (122) and (123). But this form is not presented in this work for conciseness.

C. Iterative Tradeoff Filter

Generally, the tradeoff filter is more flexible in controlling the compromise between noise reduction and speech distortion. In this subsection, we derive the iterative tradeoff filter by solving the following problems:

$$\min_{\underline{\mathbf{h}}_{\mathrm{S},P}^{(n)}} J_{\mathrm{d}}(\underline{\mathbf{h}}_{\mathrm{S},P}^{(n)}|\underline{\mathbf{h}}_{\mathrm{T},P}^{(n-1)},\underline{\mathbf{h}}_{\mathrm{F},P}^{(n-1)}) \quad \text{s. t. } J_{\mathrm{r}}(\underline{\mathbf{h}}_{\mathrm{S},P}^{(n)}|\underline{\mathbf{h}}_{\mathrm{T},P}^{(n-1)},\underline{\mathbf{h}}_{\mathrm{F},P}^{(n-1)}) = \alpha_1\sigma_{V_1}^2, \tag{87}$$

$$\min_{\underline{\mathbf{h}}_{\mathrm{T},P}^{(n)}} J_{\mathrm{d}}(\underline{\mathbf{h}}_{\mathrm{T},P}^{(n)}|\underline{\mathbf{h}}_{\mathrm{S},P}^{(n)},\underline{\mathbf{h}}_{\mathrm{F},P}^{(n-1)}) \quad \text{s. t. } J_{\mathrm{r}}(\underline{\mathbf{h}}_{\mathrm{T},P}^{(n)}|\underline{\mathbf{h}}_{\mathrm{S},P}^{(n)},\underline{\mathbf{h}}_{\mathrm{F},P}^{(n-1)}) = \alpha_2\sigma_{V_1}^2, \tag{88}$$

$$\min_{\underline{\mathbf{h}}_{\mathrm{F},P}^{(n)}} J_{\mathrm{d}}(\underline{\mathbf{h}}_{\mathrm{F},P}^{(n)}|\underline{\mathbf{h}}_{\mathrm{S},P}^{(n)},\underline{\mathbf{h}}_{\mathrm{T},P}^{(n)}) \quad \text{s. t. } J_{\mathrm{r}}(\underline{\mathbf{h}}_{\mathrm{F},P}^{(n)}|\underline{\mathbf{h}}_{\mathrm{S},P}^{(n)},\underline{\mathbf{h}}_{\mathrm{T},P}^{(n)}) = \alpha_3\sigma_{V_1}^2, \tag{89}$$

where $0 < \alpha_1,\ \alpha_2,\ \alpha_3 < 1$. Consider the initialization of $\underline{\mathbf{h}}_{\mathrm{T},P,\mu_2}^{(0)}$ as

$$\underline{\mathbf{h}}_{\mathrm{T},P,\mu_2}^{(0)} = \left[\ \mathbf{h}_{\mathrm{TTr},\mu_2,1}^T\ \ \mathbf{h}_{\mathrm{TTr},\mu_2,2}^T\ \ \cdots\ \ \mathbf{h}_{\mathrm{TTr},\mu_2,P}^T\ \right]^T, \tag{90}$$

where

$$\mathbf{h}_{\mathrm{TTr},\mu_2,p} = \frac{\phi_{X_p}^2\boldsymbol{\Phi}_{\mathrm{in,T},p}^{-1}\boldsymbol{\rho}_{\mathbf{x}_{\mathrm{T},p}X_p}}{\mu_2 + \phi_{X_p}^2\boldsymbol{\rho}_{\mathbf{x}_{\mathrm{T},p}X_p}^H\boldsymbol{\Phi}_{\mathrm{in,T},p}^{-1}\boldsymbol{\rho}_{\mathbf{x}_{\mathrm{T},p}X_p}}, \quad p = 1,2,\ldots,P, \tag{91}$$

and the initialization of $\underline{\mathbf{h}}_{\mathrm{F},P,\mu_3}^{(0)}$ as

$$\underline{\mathbf{h}}_{\mathrm{F},P,\mu_3}^{(0)} = \left[\ \mathbf{h}_{\mathrm{FTr},\mu_3,1}^T\ \ \mathbf{h}_{\mathrm{FTr},\mu_3,2}^T\ \ \cdots\ \ \mathbf{h}_{\mathrm{FTr},\mu_3,P}^T\ \right]^T, \tag{92}$$

where

$$\mathbf{h}_{\mathrm{FTr},\mu_3,p} = \frac{\phi_{X_p}^2\boldsymbol{\Phi}_{\mathrm{in,F},p}^{-1}\boldsymbol{\rho}_{\mathbf{x}_{\mathrm{F},p}X_p}}{\mu_3 + \phi_{X_p}^2\boldsymbol{\rho}_{\mathbf{x}_{\mathrm{F},p}X_p}^H\boldsymbol{\Phi}_{\mathrm{in,F},p}^{-1}\boldsymbol{\rho}_{\mathbf{x}_{\mathrm{F},p}X_p}}, \quad p = 1,2,\ldots,P. \tag{93}$$

Note that $\mathbf{h}_{\mathrm{TTr},\mu_2,p},\ p = 1,\ 2,\ \ldots,\ P$, is the tradeoff filter of length N of the pth channel when only the inter-frame information is taken into account, and $\mathbf{h}_{\mathrm{FTr},\mu_3,p},\ p = 1,\ 2,\ \ldots,\ P$, is the tradeoff filter of length L of the pth channel when only the inter-band information is taken into account [29].

Then, iteratively solving the cost functions (87), (88) and (89) yields the iterative tradeoff sub-filters

$$\underline{\mathbf{h}}_{\mathrm{S},P,\mu_1}^{(n)} = \frac{\phi_{X_1}^2\left(\boldsymbol{\Phi}_{\mathrm{in,TF},P}^{(n-1)}\right)^{-1}\underline{\boldsymbol{\rho}}_{\underline{\mathbf{x}}X_1,\mathrm{TF},P}^{(n-1)}}{\mu_1 + \phi_{X_1}^2\left(\underline{\boldsymbol{\rho}}_{\underline{\mathbf{x}}X_1,\mathrm{TF},P}^{(n-1)}\right)^H\left(\boldsymbol{\Phi}_{\mathrm{in,TF},P}^{(n-1)}\right)^{-1}\underline{\boldsymbol{\rho}}_{\underline{\mathbf{x}}X_1,\mathrm{TF},P}^{(n-1)}}, \tag{94}$$

$$\underline{\mathbf{h}}_{\mathrm{T},P,\mu_2}^{(n)} = \frac{\phi_{X_1}^2\left(\boldsymbol{\Phi}_{\mathrm{in,SF},P}^{(n)}\right)^{-1}\underline{\boldsymbol{\rho}}_{\underline{\mathbf{x}}X_1,\mathrm{SF},P}^{(n)}}{\mu_2 + \phi_{X_1}^2\left(\underline{\boldsymbol{\rho}}_{\underline{\mathbf{x}}X_1,\mathrm{SF},P}^{(n)}\right)^H\left(\boldsymbol{\Phi}_{\mathrm{in,SF},P}^{(n)}\right)^{-1}\underline{\boldsymbol{\rho}}_{\underline{\mathbf{x}}X_1,\mathrm{SF},P}^{(n)}} \tag{95}$$

and

$$\underline{\mathbf{h}}_{\mathrm{F},P,\mu_3}^{(n)} = \frac{\phi_{X_1}^2\left(\boldsymbol{\Phi}_{\mathrm{in,ST},P}^{(n)}\right)^{-1}\underline{\boldsymbol{\rho}}_{\underline{\mathbf{x}}X_1,\mathrm{ST},P}^{(n)}}{\mu_3 + \phi_{X_1}^2\left(\underline{\boldsymbol{\rho}}_{\underline{\mathbf{x}}X_1,\mathrm{ST},P}^{(n)}\right)^H\left(\boldsymbol{\Phi}_{\mathrm{in,ST},P}^{(n)}\right)^{-1}\underline{\boldsymbol{\rho}}_{\underline{\mathbf{x}}X_1,\mathrm{ST},P}^{(n)}}. \tag{96}$$

The detailed derivation steps of the above filters are provided in Appendix B.

Finally, the iterative tradeoff filter at the nth iteration is constructed as

$$\underline{\mathbf{h}}_{\mathrm{Tr},P,\mu_1,\mu_2,\mu_3} = \sum_{p=1}^{P}\mathbf{h}_{\mathrm{F},p,\mu_3}^{(n)} \otimes \mathbf{h}_{\mathrm{T},p,\mu_2}^{(n)} \otimes \mathbf{h}_{\mathrm{S},p,\mu_1}^{(n)}. \tag{97}$$

Depending on the values of μ_1, μ_2 and μ_3, we have the following four cases:

- $\mu_1 = \mu_2 = \mu_3 = 1$, $\underline{\mathbf{h}}_{\mathrm{Tr},P,\mu_1,\mu_2,\mu_3} = \underline{\mathbf{h}}_{\mathrm{W},P}$ is the iterative Wiener filter;
- $\mu_1 = \mu_2 = \mu_3 = 0$, $\underline{\mathbf{h}}_{\mathrm{Tr},P,\mu_1,\mu_2,\mu_3}$ degenerates to the iterative MVDR filter $\underline{\mathbf{h}}_{\mathrm{MVDR},P}$, though the iterative MVDR filter cannot be deduced directly from the derivation process of $\underline{\mathbf{h}}_{\mathrm{Tr},P,\mu_1,\mu_2,\mu_3}$;
- $\mu_1,\ \mu_2,\ \mu_3 > 1$, $\underline{\mathbf{h}}_{\mathrm{Tr},P,\mu_1,\mu_2,\mu_3}$ is a filter with lower residual noise and higher speech distortion than $\underline{\mathbf{h}}_{\mathrm{W},P}$;
- $\mu_1,\ \mu_2,\ \mu_3 < 1$, $\underline{\mathbf{h}}_{\mathrm{Tr},P,\mu_1,\mu_2,\mu_3}$ is a filter with higher residual noise and lower speech distortion than $\underline{\mathbf{h}}_{\mathrm{W},P}$.

V. Computational Complexity

If it is performed using the Gauss-Jordan elimination method [53], the inversion of a real-valued matrix involves $3 \times Q^3$ real-valued multiplications, where $Q \times Q$ is the size of the matrix needed to be inverted. The computational complexity (where only the number of real-valued multiplications is considered) of the iterative Wiener filter is summarized in Table I as an example (the computational complexity of other iterative filters can be obtained analogously). Note that there is no tensor decomposition involved in the proposed noise reduction algorithms. As known, the conventional STFT domain multichannel Wiener filter is $\underline{\mathbf{h}}_{\mathrm{CW}} = \phi_{X_1}^2\boldsymbol{\Phi}_{\underline{\mathbf{y}}}^{-1}\underline{\boldsymbol{\rho}}_{\underline{\mathbf{x}}X_1}$ [29]. The number of real-valued multiplications needed for computing $\underline{\mathbf{h}}_{\mathrm{CW}}$ is $4 \times [3 \times (MNL)^3 + (MNL)^2]$.

Fig. 2 plots the computational complexity of the iterative Wiener filter (for steps estimating the desired speech at a single frame n and a single frequency bin k) as a function of the iteration number, O, and the microphone number, M, for different values of P. For comparison, we also plot the computational complexity of the traditional Wiener filter. As seen from Figs. 2 a and 2 b, the computational complexity of the iterative Wiener filter is significantly lower than that of the traditional Wiener filter as long as the values of P and O are small, especially when signal dimensions M, N and L are large. In addition, as observed from Fig. 2 c, the gain in computational complexity saving of the proposed iterative Wiener filter compared with the conventional Wiener filter increases with the increase of the microphone number, M. Note that the iterative filters converge within only

TABLE I
COMPUTATIONAL COMPLEXITY OF THE ITERATIVE WIENER FILTER AS A FUNCTION OF THE MICROPHONE NUMBER M, PARAMETERS N, L AND P, AND THE ITERATION NUMBER O

Algorithm step	Real-Valued Multiplications
Computing $\underline{\mathbf{h}}_{\mathrm{T},P}^{(0)}$	$P\times 4\times(3\times N^3+N^2)$
Computing $\underline{\mathbf{h}}_{\mathrm{F},P}^{(0)}$	$P\times 4\times(3\times L^3+L^2)$
Computing $\underline{\mathbf{H}}_{\mathrm{TF},P}^{(n-1)}$	$4\times NL\times P$
Computing $\underline{\mathbf{H}}_{\mathrm{TF},P}^{(n-1)}\boldsymbol{\Phi}_{\underline{\mathbf{y}}}\left(\underline{\mathbf{H}}_{\mathrm{TF},P}^{(n-1)}\right)^H$	$4\times(NL\times MP\times MNL+NL\times MP\times MP)$
Computing $\underline{\mathbf{H}}_{\mathrm{TF},P}^{(n-1)}\boldsymbol{\rho}_{\underline{\mathbf{x}}X_1}$	$4\times NL\times MP$
Computing $\underline{\mathbf{h}}_{\mathrm{S},P}^{(n)}$	$4\times[3\times(MP)^3+(MP)^2]$
Computing $\underline{\mathbf{H}}_{\mathrm{SF},P}^{(n)}$	$4\times ML\times P$
Computing $\underline{\mathbf{H}}_{\mathrm{SF},P}^{(n)}\boldsymbol{\Phi}_{\underline{\mathbf{y}}}\left(\underline{\mathbf{H}}_{\mathrm{SF},P}^{(n)}\right)^H$	$4\times(ML\times NP\times MNL+ML\times NP\times NP)$
Computing $\underline{\mathbf{H}}_{\mathrm{SF},P}^{(n)}\boldsymbol{\rho}_{\underline{\mathbf{x}}X_1}$	$4\times ML\times NP$
Computing $\underline{\mathbf{h}}_{\mathrm{T},P}^{(n)}$	$4\times[3\times(NP)^3+(NP)^2]$
Computing $\underline{\mathbf{H}}_{\mathrm{ST},P}^{(n)}$	$4\times MN\times P$
Computing $\underline{\mathbf{H}}_{\mathrm{ST},P}^{(n)}\boldsymbol{\Phi}_{\underline{\mathbf{y}}}\left(\underline{\mathbf{H}}_{\mathrm{ST},P}^{(n)}\right)^H$	$4\times(MN\times LP\times MNL+MN\times LP\times LP)$
Computing $\underline{\mathbf{H}}_{\mathrm{SF},P}^{(n)}\boldsymbol{\rho}_{\underline{\mathbf{x}}X_1}$	$4\times MN\times LP$
Computing $\underline{\mathbf{h}}_{\mathrm{F},P}^{(n)}$	$4\times[3\times(LP)^3+(LP)^2]$
Computing $\underline{\mathbf{h}}_{\mathrm{W},P}$	$4\times P\times M\times N\times L$
Total	$4\times[P\times(3\times N^3+N^2)+P\times(3\times L^3+L^2)]+O\times 4\times[NL\times P+NL\times MP\times MNL+NL\times MP\times MP+NL\times MP+3\times(MP)^3+(MP)^2+ML\times P+ML\times NP\times MNL+ML\times NP\times NP+ML\times NP+3\times(NP)^3+(NP)^2+MN\times P+MN\times LP\times MNL+MN\times LP\times LP+MN\times LP+3\times(LP)^3+(LP)^2]+4\times P\times M\times N\times L$

a small number of iterations, which will be discussed in the next section.

VI. SIMULATIONS

In this section, we investigate the performance of the algorithms developed in Section IV through simulations and compare the results with those of the conventional filters.

A. Simulation Setup

Simulations were conducted using the room impulse responses measured in the varechoic chamber at Bell Labs. The detailed information of the layout of the chamber, arrangement of the microphones and loudspeakers, electronic panels, sampling rate, and data bits can be found in [30], [54]. In our simulations, one loudspeaker at the position $(1.337\mathrm{m}, 1.938\mathrm{m}, 1.600\mathrm{m})$ was used as the clean speech source. A uniform linear microphone array of 6 microphones was configured to record signals at a sampling rate of 16kHz and the 6 microphones, respectively, are at positions $(x, 0.500\mathrm{m}, 1.400\mathrm{m})$, where $x = 2.437 : 0.1 : 2.937$ (in meter). The reverberation time, T_{60}, which is defined as the time for the sound to die away

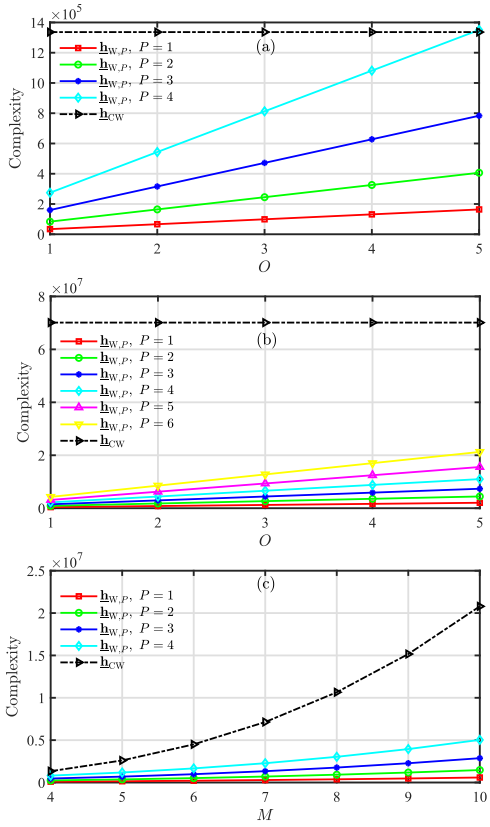


Fig. 2. Computational complexity of the iterative and conventional Wiener filters as a function of the parameter O and M for different values of P. Conditions of (a): $M = 4$, $N = 4$ and $L = 3$, conditions of (b): $M = 6$, $N = 6$ and $L = 5$, conditions of (c): $N = 4$, $L = 3$ and $O = 3$.

to a level of 60 dB below its original level, was controlled to be approximately 240 ms and 580 ms. All room impulse responses from the source position to the microphone positions were measured and the output signals of the microphones were generated by convolving the clean source signal with the corresponding measured impulse responses, and then added together with noise to control the iSNR level. The clean source signals were taken randomly from the TIMIT database [55], which includes speech signals of 5 male and 5 female speakers. The convolved speech signal at the 1st microphone is treated as the desired speech to recover. Three types of noise, namely, spatially and temporally white Gaussian (STWG) noise, temporally nonstationary interference (TNSI, spatially stationary except for the third simulation) noises, and nonstationary diffuse (NSD) noises [56], were used to simulate different background noise. The TNSI noise was generated by convolving the measured impulse responses from the loudspeaker positions $(x, 1.938\mathrm{m}, 1.600\mathrm{m})$, where $x = 4.337 : 1.0 : 6.337$ (in meter), to the microphone array with three pieces of keyboard typing and machine gun noise. The NSD noise field was generated by the method in [56] coupled with the babble and factory1 noises. In order to ensure the numerical stability, a small amount of the STWG noise was added into the TNSI and NSD noise fields, and the ratio of the TNSI and NSD noises to STWG noise was 20 dB. The machine

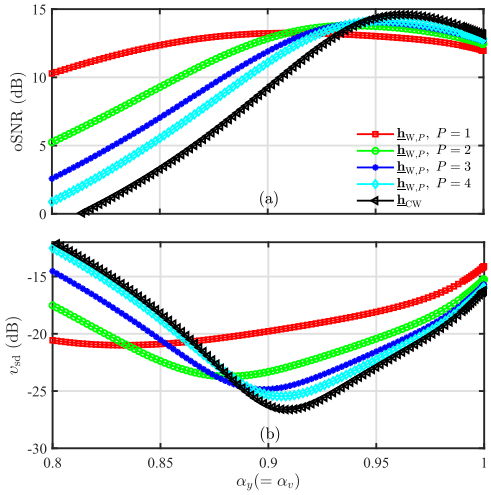


Fig. 3. Performance of the iterative Wiener filter (for different values of P) and conventional Wiener filter as functions of $\alpha_y(=\alpha_v)$ in the STWG noise field: (a) oSNR, and (b) speech distortion. Conditions: iSNR $=0$ dB, $M=4$, $N=4$, $L=3$, $O=3$, and $T_{60}\approx 240$ ms.

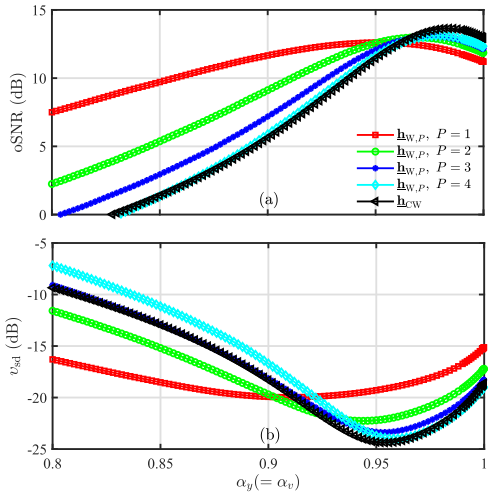


Fig. 4. Performance of the iterative Wiener filter (for different values of P) and conventional Wiener filter as functions of $\alpha_y(=\alpha_v)$ in the TNSI noise (keyboard typing noise) field: (a) oSNR, and (b) speech distortion. Conditions: iSNR $=0$ dB, $M=4$, $N=4$, $L=3$, $O=3$, and $T_{60}\approx 240$ ms.

gun, babble and factory1 noises were taken from NOISEX-92 corpus [57].

To perform noise reduction in the STFT domain, we divided the microphones' outputs into frames of 256 samples (16 ms) with 75% overlap. The Kaiser window was applied to the frames to deal with the frequency aliasing problem. Then, the fast Fourier transform (FFT) was used to each frame to realize STFT. A noise reduction filter was then designed and applied to do noise reduction. Finally, the overlap-add method was adopted for reconstructing the processed signal back into the time domain.

Implementation of the developed noise reduction algorithms requires to know the correlation matrices of the clean speech and noise signals, which can be estimated using the methods

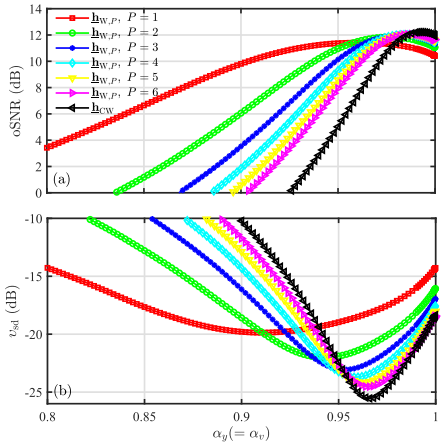


Fig. 5. Performance of the iterative Wiener filter (for different values of P) and conventional Wiener filter as functions of $\alpha_y(=\alpha_v)$ in the NSD noise (factory1 noise) field: (a) oSNR, and (b) speech distortion. Conditions: iSNR $=0$ dB, $M=6$, $N=6$, $L=5$, $O=3$, and $T_{60}\approx 580$ ms.

such as the ones in [58], [59]. However, our focus in the simulations is to examine the feasibility of the developed filters for noise reduction as well as to achieve a fair comparison with the traditional methods. In this case, we directly compute the required correlation matrices from the respective signals using the following recursive method:

$$\widehat{\mathbf{\Phi}}_{\underline{\mathbf{y}}}(k,n)=\alpha_y\widehat{\mathbf{\Phi}}_{\underline{\mathbf{y}}}(k,n-1)+(1-\alpha_y)\underline{\mathbf{y}}(k,n)\underline{\mathbf{y}}^H(k,n),$$
$$\widehat{\mathbf{\Phi}}_{\underline{\mathbf{v}}}(k,n)=\alpha_v\widehat{\mathbf{\Phi}}_{\underline{\mathbf{v}}}(k,n-1)+(1-\alpha_v)\underline{\mathbf{v}}(k,n)\underline{\mathbf{v}}^H(k,n),$$

where $0<\alpha_y,\ \alpha_v<1$ are two forgetting factors that control the influence of the previous data samples on the current estimates, and $\widehat{\mathbf{\Phi}}_{\underline{\mathbf{y}}}(k,n)$ and $\widehat{\mathbf{\Phi}}_{\underline{\mathbf{v}}}(k,n)$ are estimates of the corresponding correlation matrices using sample average. For simplicity, we set $\alpha_y=\alpha_v$ in our simulations. Then, the clean speech correlation matrix, $\widehat{\mathbf{\Phi}}_{\underline{\mathbf{x}}}(k,n)$, was computed by $\widehat{\mathbf{\Phi}}_{\underline{\mathbf{y}}}(k,n)-\widehat{\mathbf{\Phi}}_{\underline{\mathbf{v}}}(k,n)$.

The performance of the proposed algorithms was evaluated using oSNR, speech distortion and MSE as the performance measures, which have been defined in Section III. Moreover, the perceptual evaluation of speech quality (PESQ) [60] and the short-time objective intelligibility (STOI) score [61] were used to assess the overall quality and intelligibility of the enhanced speech, respectively. In terms of the PESQ, all the speech signals were used to compute the PESQ mean opinion score (MOS) in every test condition, which was then mapped to the PESQ MOS-LQO (listening quality objective) using the following mapping function [62]:

$$\mathrm{PESQ}_{\mathrm{MOS-LQO}}=0.999+\frac{4}{1+e^{-1.4945\times\mathrm{PESQ}_{\mathrm{MOS}}+4.6607}}. \tag{98}$$

The range of PESQ MOS-LQO score is between 1.02 and 4.55, and the STOI score ranges from 0 to 1. Higher PESQ MOS-LQO and STOI scores imply better speech quality and intelligibility, respectively.

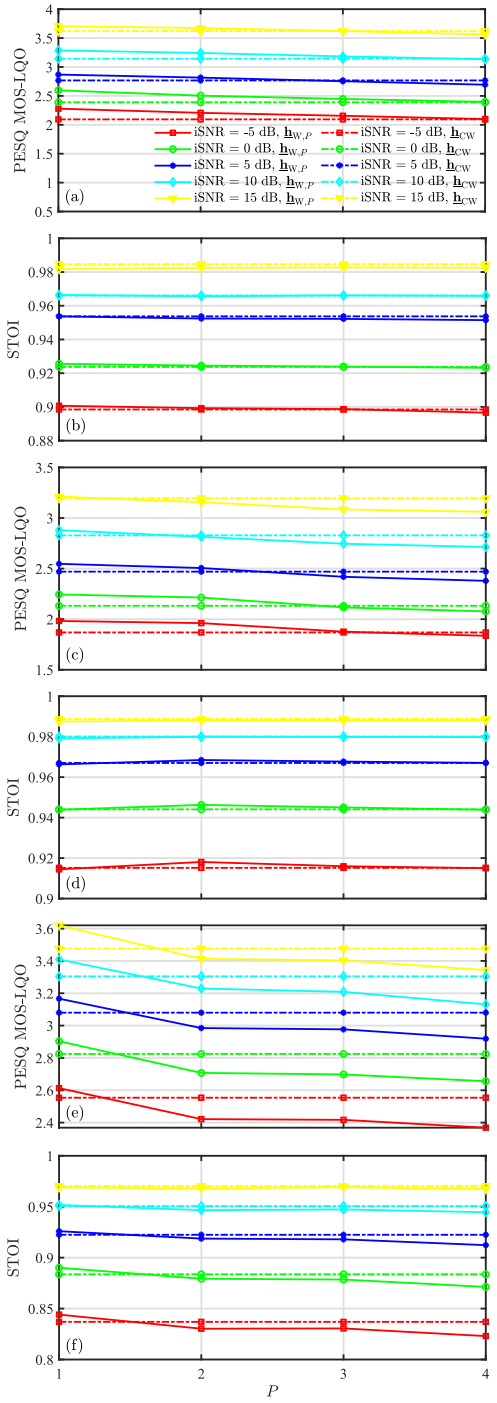


Fig. 6. Performance of the iterative Wiener filter (as functions of P) and conventional Wiener filter for different input SNRs: (a) and (b), STWG noise field, (c) and (d), TNSI (keyboard typing) noise field, (e) and (f), TNSI (machine gun) noise field. Conditions: $M = 4$, $N = 4$, $L = 3$, $O = 3$, and $T_{60} \approx 240$ ms.

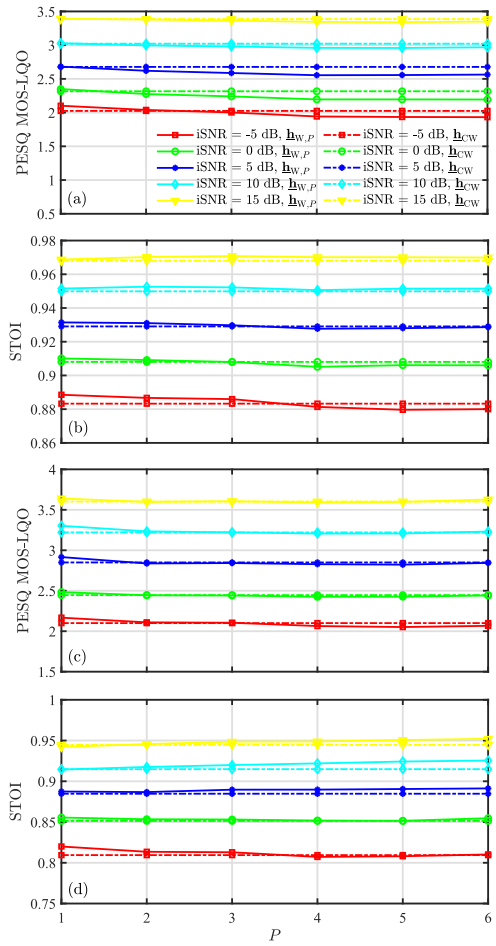


Fig. 7. Performance of the iterative Wiener filter (as functions of P) and conventional Wiener filter for different input SNRs: (a) and (b), NSD (babble) noise field, (c) and (d), NSD (factory1) noise field. Conditions: $M = 6$, $N = 6$, $L = 5$, $O = 3$, and $T_{60} \approx 580$ ms.

B. Simulation Results

In the first simulation, the effect of the forgetting factor on the performance of the iterative Wiener filter is evaluated in the STWG, TNSI (keyboard typing noise) and NSD (factory1 noise) noise fields. In the STWG and TNSI noise fields, we set the number of microphones M to 4, the parameters N to 4, L to 3 ($K_k^- = K_k^+ = 1$, the same hereinafter) and $T_{60} \approx 240$ ms, and in the NSD noise field, we set the number of microphones M to 6, the parameters N to 6, L to 5 ($K_k^- = K_k^+ = 2$, the same hereinafter) and $T_{60} \approx 580$ ms. In all three cases, we set the iteration number O to 3, and the input SNR to 0 dB. Figures 4, 6 and 6 plot the oSNR and speech distortion index of the iterative Wiener filter as functions of the forgetting factor for different values of P, respectively, in the STWG, TNSI and NSD noise fields. For comparison, the results of the traditional Wiener filter are also plotted. As seen, in the three different conditions, the oSNR of the iterative Wiener filter with different values of P and the traditional Wiener filter first increases with the increase of the forgetting factor, and then decreases. In contrast, the speech distortion index first decreases with the increase of the forgetting factor, and then increases. If the value of the forgetting factor that gives the maximum oSNR is treated as the optimal forgetting factor, the optimal forgetting factor of the iterative Wiener filter increases with the increase of P, and is less than that of the traditional Wiener filter. This indicates indirectly that, the smaller the parameter P, the less observed samples are needed in estimating the signal statistics for computing the

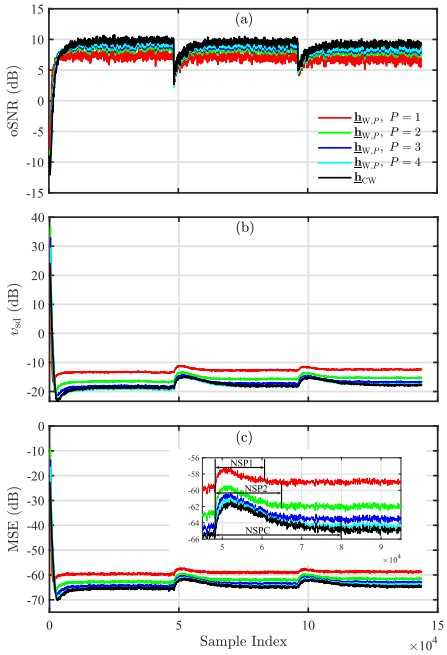


Fig. 8. Performance of the iterative and conventional Wiener filters in the spatial nonstationary interference noise (white Gaussian point sources) field for different values of P: (a) oSNR, (b) speech distortion, and (c) MSE. Conditions: iSNR $= 0$ dB, $M = 4$, $N = 4$, $L = 3$, $O = 3$, and $T_{60} \approx 240$ ms.

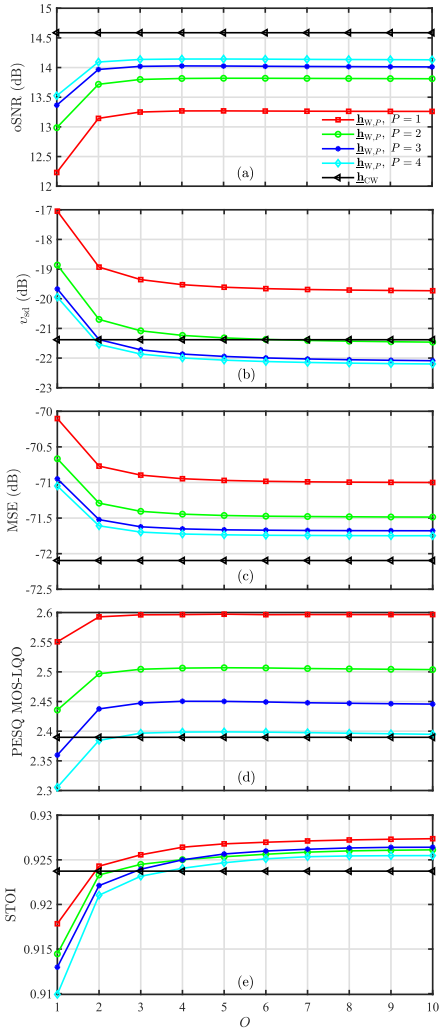


Fig. 9. Performance of the iterative Wiener filter as functions of the iteration number, O, in the STWG noise field for different values of P: (a) oSNR, (b) speech distortion, (c) MSE, (d) PESQ score, and (e) STOI score. Conditions: iSNR $= 0$ dB, $M = 4$, $N = 4$, $L = 3$, and $T_{60} \approx 240$ ms.

iterative Wiener filter, and thus the iterative Wiener filter has better tracking ability for the temporal/spatial nonstationarity of the speech and noise signals than the traditional Wiener filter, especially when P is small. We can also observe that in all three simulated cases, with the optimal forgetting factor, the oSNR and speech distortion index of the iterative Wiener filter are comparable with those of the traditional Wiener filter when P is relatively large. Note that we use the optimal forgetting factor for each specific noise and T_{60} condition in the rest of the simulations.

Next, we investigate the PESQ and STOI of the speech enhanced by the iterative Wiener filter as functions of the parameter P under different iSNR levels. Again, in the STWG and TNSI noise fields, we set the number of microphones M to 4, the parameters N to 4, L to 3 and $T_{60} \approx 240$ ms. In the NSD noise field, we set the number of microphones M to 6, the parameters N to 6, L to 5 and $T_{60} \approx 580$ ms. In all simulated cases, we set O to 3. The results are plotted in Figs. 7 and 8, which also include the results of the conventional Wiener filter. As seen, in most of the simulated conditions, we can obtain comparable or even higher PESQ and STOI score by using the iterative Wiener filter compared with the traditional Wiener filter if the parameter P is appropriately selected, especially for small P which corresponds to significantly lower computational complexity. The underlying reason of this phenomenon is that, though not optimal, the iterative Wiener filter tracks the temporal/spatial signal nonstationarity better.

The third simulation investigates the tracking ability of the proposed iterative and conventional Wiener filters with respect to the temporal/spatial signal nonstationarity. Considering that the nonstationarity of speech signal may lead to significant segmental performance measure fluctuation and make it difficult to clearly reveal the actual tracking performance, we used 20 segments of white Gaussian noise as the clean signal with length of 9 seconds. Three white Gaussian point interferences with length of 3 seconds eradiating from the locations (4.337 m, 1.938 m, 1.600 m), (5.337 m, 1.938 m, 1.600 m) and (6.337 m, 1.938 m, 1.600 m) were mixed sequentially with the clean signal. The STWG noise was also mixed with the clean signal to ensure the numerical stability. The ratio of the interference to STWG noise was 20 dB. We set the input SNR to 0 dB, and the parameters M to 4, N to 4, L to 3, O to 3, and $T_{60} \approx 240$ ms. The averaged simulation results are presented in Fig. 8, where 'NSP1,' 'NSP2' and 'NSPC' stand for nonstationary period of the filters $\underline{\mathbf{h}}_{\mathrm{W},P}(P = 1)$, $\underline{\mathbf{h}}_{\mathrm{W},P}(P = 2)$ and $\underline{\mathbf{h}}_{\mathrm{CW}}$, respectively. As can be seen from the magnified picture of the second part of Fig. 8 c, i.e., the MSEs of the proposed iterative and conventional Wiener filters, the former with small values of P converges obviously faster than the later when the interference location suddenly changes.

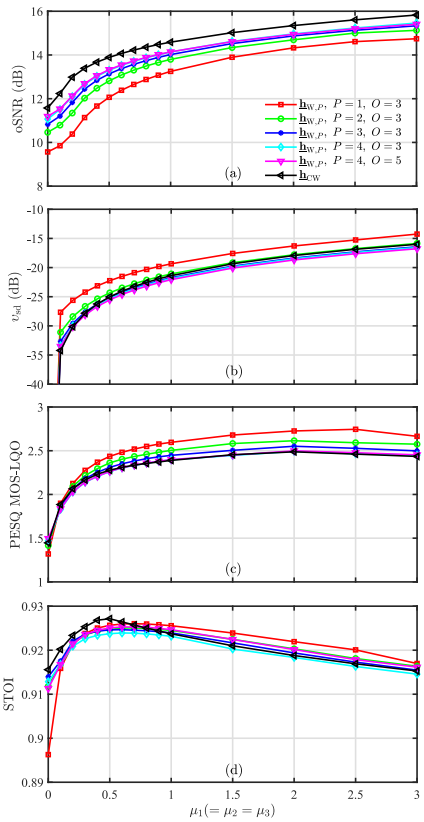


Fig. 10. Performance of the iterative tradeoff filter as functions of $\mu_1 (= \mu_2 = \mu_3)$ in the STWG noise field for different values of P: (a) oSNR, (b) speech distortion, (c) PESQ score, and (d) STOI score. Conditions: iSNR $= 0$ dB, $M = 4$, $N = 4$, $L = 3$, and $T_{60} \approx 240$ ms.

This indicates the improved tracking ability of the iterative filters with respect to the temporal/spatial signal nonstationarity.

The fourth simulation investigates the effect of the iteration number on the performance of the iterative Wiener filter. The noise considered is the STWG noise, and we set the input SNR to 0 dB, M to 4, N to 4, L to 3, and $T_{60} \approx 240$ ms. The performance of the iterative Wiener filter versus the iteration number is presented in Fig. 9, which also includes the results of the conventional Wiener filter for comparison. One can see that, for different values of P, the oSNR and PESQ score increase first with the increase of the iteration number, O, and then decrease slightly. The speech distortion index and MSE decrease with the increase of the iteration number, and the STOI score of the enhanced speech increases with the increase of O. It is observed that the iterative Wiener filter converges within only 3 to 4 iterations for all values of P.

At last, we evaluate the effect of the parameters μ_1, μ_2 and μ_3 on the performance of the iterative tradeoff filter. Again, we set the input SNR to 0 dB, M to 4, N to 4, L to 3, O to 3, $T_{60} \approx 240$ ms, and set $\mu_1 = \mu_2 = \mu_3$ for simplicity. For comparison, the result of the conventional Wiener filter and the iterative Wiener filter with $P = 4$ and $O = 5$ are also included. The results are presented in Fig. 10. As we can see, the oSNR and speech distortion of the iterative MVDR filter ($\mu_1 = 0$) is the lowest for all values of P. Both the oSNR and speech distortion increase with the increase of the parameter μ_1 for all values of P, which coincides with the analysis above. The PESQ and STOI, however, first increase with μ_1, and then decrease. It can be seen that, in most cases, we obtain the highest PESQ score when $P = 1$. For the STOI, however, the conventional Wiener filter has better performance when $\mu < 0.7$, and the iterative Wiener filter has better performance when $\mu \geq 0.7$.

VII. Conclusion

In this work, we have introduced a class of multichannel iterative filters in the STFT domain, which exploit the Kronecker product decomposition for separating the spatial, temporal and frequency roles of the noise reduction filter. Instead of computing a long noise reduction filter, we derive three much shorter sub-filters, which leads to two advantages over the traditional algorithms: 1) significantly lower computational complexity; 2) better tracking ability for the temporal/spatial signal nonstationarity. The preliminary simulation results show that the proposed iterative filters can achieve similar or even better noise reduction performance with significantly lower computational complexity compared with the conventional algorithms if the parameter is appropriately selected. The proposed framework can be combined with fast matrix inversion algorithms (which of course are also available and useful for conventional approaches) straightforwardly to further lower the computational complexity.

Appendix A

Using $\underline{\mathbf{h}}_{\mathrm{T},P}^{(0)}$ and $\underline{\mathbf{h}}_{\mathrm{F},P}^{(0)}$ to construct $\underline{\mathbf{H}}_{\mathrm{TF},P}^{(0)}$, and substituting it into (55) and (56), we obtain

$$\underline{\mathbf{\Phi}}_{\underline{\mathbf{y}}_{\mathrm{TF},P}}^{(0)} = \underline{\mathbf{H}}_{\mathrm{TF},P}^{(0)} \underline{\mathbf{\Phi}}_{\underline{\mathbf{y}}} \left(\underline{\mathbf{H}}_{\mathrm{TF},P}^{(0)} \right)^H, \tag{99}$$

$$\underline{\boldsymbol{\rho}}_{\underline{\mathbf{x}}X_1,\mathrm{TF},P}^{(0)} = \underline{\mathbf{H}}_{\mathrm{TF},P}^{(0)} \underline{\boldsymbol{\rho}}_{\underline{\mathbf{x}}X_1}. \tag{100}$$

Substituting (99) and (100) into (61) yields

$$J(\underline{\mathbf{h}}_{\mathrm{S},P}^{(1)} | \underline{\mathbf{h}}_{\mathrm{T},P}^{(0)}, \underline{\mathbf{h}}_{\mathrm{F},P}^{(0)}) = \phi_{X_1}^2 - 2\phi_{X_1}^2 \left(\underline{\mathbf{h}}_{\mathrm{S},P}^{(1)} \right)^H \underline{\boldsymbol{\rho}}_{\underline{\mathbf{x}}X_1,\mathrm{TF},P}^{(0)} + \left(\underline{\mathbf{h}}_{\mathrm{S},P}^{(1)} \right)^H \underline{\mathbf{\Phi}}_{\underline{\mathbf{y}}_{\mathrm{TF},P}}^{(0)} \underline{\mathbf{h}}_{\mathrm{S},P}^{(1)}. \tag{101}$$

Finding the gradient of (101) with respect to $\underline{\mathbf{h}}_{\mathrm{S},P}^{(1)}$ and force the result to 0, we obtain

$$\underline{\mathbf{h}}_{\mathrm{S},P}^{(1)} = \phi_{X_1}^2 \left(\underline{\mathbf{\Phi}}_{\underline{\mathbf{y}}_{\mathrm{TF},P}}^{(0)} \right)^{-1} \underline{\boldsymbol{\rho}}_{\underline{\mathbf{x}}X_1,\mathrm{TF},P}^{(0)}. \tag{102}$$

Using $\underline{\mathbf{h}}_{\mathrm{S},P}^{(1)}$ and $\underline{\mathbf{h}}_{\mathrm{F},P}^{(0)}$ to construct $\underline{\mathbf{H}}_{\mathrm{SF},P}^{(1)}$, and substituting it into (57) and (58), we obtain

$$\underline{\mathbf{\Phi}}_{\underline{\mathbf{y}}_{\mathrm{SF},P}}^{(1)} = \underline{\mathbf{H}}_{\mathrm{SF},P}^{(1)} \underline{\mathbf{\Phi}}_{\underline{\mathbf{y}}} \left(\underline{\mathbf{H}}_{\mathrm{SF},P}^{(1)} \right)^H, \tag{103}$$

$$\underline{\boldsymbol{\rho}}_{\underline{\mathbf{x}}X_1,\mathrm{SF},P}^{(1)} = \underline{\mathbf{H}}_{\mathrm{SF},P}^{(1)} \underline{\boldsymbol{\rho}}_{\underline{\mathbf{x}}X_1}. \tag{104}$$

With $\underline{\mathbf{\Phi}}_{\underline{\mathbf{y}}_{\mathrm{SF},P}}^{(1)}$ and $\underline{\boldsymbol{\rho}}_{\underline{\mathbf{x}}X_1,\mathrm{SF},P}^{(1)}$, (62) can be written as

$$J(\underline{\mathbf{h}}_{\mathrm{T},P}^{(1)} | \underline{\mathbf{h}}_{\mathrm{S},P}^{(1)}, \underline{\mathbf{h}}_{\mathrm{F},P}^{(0)}) = \phi_{X_1}^2 - 2\phi_{X_1}^2 \left(\underline{\mathbf{h}}_{\mathrm{T},P}^{(1)} \right)^H \underline{\boldsymbol{\rho}}_{\underline{\mathbf{x}}X_1,\mathrm{SF},P}^{(1)} + \left(\underline{\mathbf{h}}_{\mathrm{T},P}^{(1)} \right)^H \underline{\mathbf{\Phi}}_{\underline{\mathbf{y}}_{\mathrm{SF},P}}^{(1)} \underline{\mathbf{h}}_{\mathrm{T},P}^{(1)}. \tag{105}$$

Minimizing (105) with respect to $\underline{\mathbf{h}}_{\mathrm{T},P}^{(1)}$ leads to

$$\underline{\mathbf{h}}_{\mathrm{T},P}^{(1)} = \phi_{X_1}^2 \left(\underline{\mathbf{\Phi}}_{\underline{\mathbf{y}}_{\mathrm{SF},P}}^{(1)}\right)^{-1} \underline{\boldsymbol{\rho}}_{\underline{\mathbf{x}}X_1,\mathrm{SF},P}^{(1)}. \tag{106}$$

Using $\underline{\mathbf{h}}_{\mathrm{S},P}^{(1)}$ and $\underline{\mathbf{h}}_{\mathrm{T},P}^{(1)}$ to construct $\underline{\mathbf{H}}_{\mathrm{ST},P}^{(1)}$, and substituting it into (59) and (60), we obtain

$$\underline{\mathbf{\Phi}}_{\underline{\mathbf{y}}_{\mathrm{ST},P}}^{(1)} = \underline{\mathbf{H}}_{\mathrm{ST},P}^{(1)} \underline{\mathbf{\Phi}}_{\underline{\mathbf{y}}} \left(\underline{\mathbf{H}}_{\mathrm{ST},P}^{(1)}\right)^H, \tag{107}$$

$$\underline{\boldsymbol{\rho}}_{\underline{\mathbf{x}}X_1,\mathrm{ST},P}^{(1)} = \underline{\mathbf{H}}_{\mathrm{ST},P}^{(1)} \underline{\boldsymbol{\rho}}_{\underline{\mathbf{x}}X_1}. \tag{108}$$

With $\underline{\mathbf{\Phi}}_{\underline{\mathbf{y}}_{\mathrm{ST},P}}^{(1)}$ and $\underline{\boldsymbol{\rho}}_{\underline{\mathbf{x}}X_1,\mathrm{ST},P}^{(1)}$, (63) can be written as

$$J(\underline{\mathbf{h}}_{\mathrm{F},P}^{(1)}|\underline{\mathbf{h}}_{\mathrm{S},P}^{(1)}, \underline{\mathbf{h}}_{\mathrm{T},P}^{(1)}) = \phi_{X_1}^2 - 2\phi_{X_1}^2 \left(\underline{\mathbf{h}}_{\mathrm{F},P}^{(1)}\right)^H \underline{\boldsymbol{\rho}}_{\underline{\mathbf{x}}X_1,\mathrm{ST},P}^{(1)} + \left(\underline{\mathbf{h}}_{\mathrm{F},P}^{(1)}\right)^H \underline{\mathbf{\Phi}}_{\underline{\mathbf{y}}_{\mathrm{ST},P}}^{(1)} \underline{\mathbf{h}}_{\mathrm{F},P}^{(1)}. \tag{109}$$

Minimizing (109) with respect to $\underline{\mathbf{h}}_{\mathrm{F},P}^{(1)}$ leads to

$$\underline{\mathbf{h}}_{\mathrm{F},P}^{(1)} = \phi_{X_1}^2 \left(\underline{\mathbf{\Phi}}_{\underline{\mathbf{y}}_{\mathrm{ST},P}}^{(1)}\right)^{-1} \underline{\boldsymbol{\rho}}_{\underline{\mathbf{x}}X_1,\mathrm{ST},P}^{(1)}. \tag{110}$$

Continuing the iterations up to iteration n, we have

$$\underline{\mathbf{h}}_{\mathrm{S},P}^{(n)} = \phi_{X_1}^2 \left(\underline{\mathbf{\Phi}}_{\underline{\mathbf{y}}_{\mathrm{TF},P}}^{(n-1)}\right)^{-1} \underline{\boldsymbol{\rho}}_{\underline{\mathbf{x}}X_1,\mathrm{TF},P}^{(n-1)}, \tag{111}$$

$$\underline{\mathbf{h}}_{\mathrm{T},P}^{(n)} = \phi_{X_1}^2 \left(\underline{\mathbf{\Phi}}_{\underline{\mathbf{y}}_{\mathrm{SF},P}}^{(n)}\right)^{-1} \underline{\boldsymbol{\rho}}_{\underline{\mathbf{x}}X_1,\mathrm{SF},P}^{(n)}, \tag{112}$$

$$\underline{\mathbf{h}}_{\mathrm{F},P}^{(n)} = \phi_{X_1}^2 \left(\underline{\mathbf{\Phi}}_{\underline{\mathbf{y}}_{\mathrm{ST},P}}^{(n)}\right)^{-1} \underline{\boldsymbol{\rho}}_{\underline{\mathbf{x}}X_1,\mathrm{ST},P}^{(n)}, \tag{113}$$

where

$$\underline{\mathbf{\Phi}}_{\underline{\mathbf{y}}_{\mathrm{TF},P}}^{(n-1)} = \underline{\mathbf{H}}_{\mathrm{TF},P}^{(n-1)} \underline{\mathbf{\Phi}}_{\underline{\mathbf{y}}} \left(\underline{\mathbf{H}}_{\mathrm{TF},P}^{(n-1)}\right)^H, \tag{114}$$

$$\underline{\boldsymbol{\rho}}_{\underline{\mathbf{x}}X_1,\mathrm{TF},P}^{(n-1)} = \underline{\mathbf{H}}_{\mathrm{TF},P}^{(n-1)} \underline{\boldsymbol{\rho}}_{\underline{\mathbf{x}}X_1}, \tag{115}$$

$$\underline{\mathbf{\Phi}}_{\underline{\mathbf{y}}_{\mathrm{SF},P}}^{(n)} = \underline{\mathbf{H}}_{\mathrm{SF},P}^{(n)} \underline{\mathbf{\Phi}}_{\underline{\mathbf{y}}} \left(\underline{\mathbf{H}}_{\mathrm{SF},P}^{(n)}\right)^H, \tag{116}$$

$$\underline{\boldsymbol{\rho}}_{\underline{\mathbf{x}}X_1,\mathrm{SF},P}^{(n)} = \underline{\mathbf{H}}_{\mathrm{SF},P}^{(n)} \underline{\boldsymbol{\rho}}_{\underline{\mathbf{x}}X_1}, \tag{117}$$

$$\underline{\mathbf{\Phi}}_{\underline{\mathbf{y}}_{\mathrm{ST},P}}^{(n)} = \underline{\mathbf{H}}_{\mathrm{ST},P}^{(n)} \underline{\mathbf{\Phi}}_{\underline{\mathbf{y}}} \left(\underline{\mathbf{H}}_{\mathrm{ST},P}^{(n)}\right)^H, \tag{118}$$

$$\underline{\boldsymbol{\rho}}_{\underline{\mathbf{x}}X_1,\mathrm{ST},P}^{(n)} = \underline{\mathbf{H}}_{\mathrm{ST},P}^{(n)} \underline{\boldsymbol{\rho}}_{\underline{\mathbf{x}}X_1}. \tag{119}$$

Since

$$\underline{\mathbf{\Phi}}_{\underline{\mathbf{y}}} = \phi_{X_1}^2 \underline{\boldsymbol{\rho}}_{\underline{\mathbf{x}}X_1} \underline{\boldsymbol{\rho}}_{\underline{\mathbf{x}}X_1}^H + \mathbf{\Phi}_{\mathrm{in}}, \tag{120}$$

the matrices $\underline{\mathbf{\Phi}}_{\underline{\mathbf{y}}_{\mathrm{TF},P}}^{(n-1)}$, $\underline{\mathbf{\Phi}}_{\underline{\mathbf{y}}_{\mathrm{SF},P}}^{(n)}$ and $\underline{\mathbf{\Phi}}_{\underline{\mathbf{y}}_{\mathrm{ST},P}}^{(n)}$ can be written as

$$\underline{\mathbf{\Phi}}_{\underline{\mathbf{y}}_{\mathrm{TF},P}}^{(n-1)} = \phi_{X_1}^2 \underline{\boldsymbol{\rho}}_{\underline{\mathbf{x}}X_1,\mathrm{TF},P}^{(n-1)} \left(\underline{\boldsymbol{\rho}}_{\underline{\mathbf{x}}X_1,\mathrm{TF},P}^{(n-1)}\right)^H + \mathbf{\Phi}_{\mathrm{in,TF},P}^{(n-1)}, \tag{121}$$

$$\underline{\mathbf{\Phi}}_{\underline{\mathbf{y}}_{\mathrm{SF},P}}^{(n)} = \phi_{X_1}^2 \underline{\boldsymbol{\rho}}_{\underline{\mathbf{x}}X_1,\mathrm{SF},P}^{(n)} \left(\underline{\boldsymbol{\rho}}_{\underline{\mathbf{x}}X_1,\mathrm{SF},P}^{(n)}\right)^H + \mathbf{\Phi}_{\mathrm{in,SF},P}^{(n)}, \tag{122}$$

$$\underline{\mathbf{\Phi}}_{\underline{\mathbf{y}}_{\mathrm{ST},P}}^{(n)} = \phi_{X_1}^2 \underline{\boldsymbol{\rho}}_{\underline{\mathbf{x}}X_1,\mathrm{ST},P}^{(n)} \left(\underline{\boldsymbol{\rho}}_{\underline{\mathbf{x}}X_1,\mathrm{ST},P}^{(n)}\right)^H + \mathbf{\Phi}_{\mathrm{in,ST},P}^{(n)}, \tag{123}$$

respectively, where

$$\mathbf{\Phi}_{\mathrm{in,TF},P}^{(n-1)} = \underline{\mathbf{H}}_{\mathrm{TF},P}^{(n-1)} \mathbf{\Phi}_{\mathrm{in}} \left(\underline{\mathbf{H}}_{\mathrm{TF},P}^{(n-1)}\right)^H, \tag{124}$$

$$\mathbf{\Phi}_{\mathrm{in,SF},P}^{(n)} = \underline{\mathbf{H}}_{\mathrm{SF},P}^{(n)} \mathbf{\Phi}_{\mathrm{in}} \left(\underline{\mathbf{H}}_{\mathrm{SF},P}^{(n)}\right)^H, \tag{125}$$

$$\mathbf{\Phi}_{\mathrm{in,ST},P}^{(n)} = \underline{\mathbf{H}}_{\mathrm{ST},P}^{(n)} \mathbf{\Phi}_{\mathrm{in}} \left(\underline{\mathbf{H}}_{\mathrm{ST},P}^{(n)}\right)^H. \tag{126}$$

Applying the Woodbury's identity [63] to (121), (122) and (123), and substituting the results to (111), (112) and (113), we obtain (72), (73) and (74), respectively.

APPENDIX B

The MSEs $J_{\mathrm{d}}(\underline{\mathbf{h}}_{\mathrm{S},P}^{(n)}|\underline{\mathbf{h}}_{\mathrm{T},P}^{(n-1)}, \underline{\mathbf{h}}_{\mathrm{F},P}^{(n-1)})$, $J_{\mathrm{r}}(\underline{\mathbf{h}}_{\mathrm{S},P}^{(n)}|\underline{\mathbf{h}}_{\mathrm{T},P}^{(n-1)}, \underline{\mathbf{h}}_{\mathrm{T},P}^{(n-1)})$, $J_{\mathrm{d}}(\underline{\mathbf{h}}_{\mathrm{T},P}^{(n)}|\underline{\mathbf{h}}_{\mathrm{S},P}^{(n)}, \underline{\mathbf{h}}_{\mathrm{F},P}^{(n-1)})$, $J_{\mathrm{r}}(\underline{\mathbf{h}}_{\mathrm{T},P}^{(n)}|\underline{\mathbf{h}}_{\mathrm{S},P}^{(n)}, \underline{\mathbf{h}}_{\mathrm{F},P}^{(n-1)})$, $J_{\mathrm{d}}(\underline{\mathbf{h}}_{\mathrm{F},P}^{(n)}|\underline{\mathbf{h}}_{\mathrm{S},P}^{(n)}, \underline{\mathbf{h}}_{\mathrm{T},P}^{(n)})$ and $J_{\mathrm{r}}(\underline{\mathbf{h}}_{\mathrm{F},P}^{(n)}|\underline{\mathbf{h}}_{\mathrm{S},P}^{(n)}, \underline{\mathbf{h}}_{\mathrm{T},P}^{(n)})$ are defined, respectively, as

$$J_{\mathrm{d}}(\underline{\mathbf{h}}_{\mathrm{S},P}^{(n)}|\underline{\mathbf{h}}_{\mathrm{T},P}^{(n-1)}, \underline{\mathbf{h}}_{\mathrm{F},P}^{(n-1)}) = \phi_{X_1}^2 - 2\phi_{X_1}^2 \left(\underline{\mathbf{h}}_{\mathrm{S},P}^{(n)}\right)^H \times \underline{\boldsymbol{\rho}}_{\underline{\mathbf{x}}X_1,\mathrm{TF},P}^{(n-1)} + \phi_{X_1}^2 \left[\left(\underline{\mathbf{h}}_{\mathrm{S},P}^{(n)}\right)^H \underline{\boldsymbol{\rho}}_{\underline{\mathbf{x}}X_1,\mathrm{TF},P}^{(n-1)}\right]^2, \tag{127}$$

$$J_{\mathrm{r}}(\underline{\mathbf{h}}_{\mathrm{S},P}^{(n)}|\underline{\mathbf{h}}_{\mathrm{T},P}^{(n-1)}, \underline{\mathbf{h}}_{\mathrm{F},P}^{(n-1)}) = \left(\underline{\mathbf{h}}_{\mathrm{S},P}^{(n)}\right)^H \mathbf{\Phi}_{\mathrm{in,TF},P}^{(n-1)} \underline{\mathbf{h}}_{\mathrm{S},P}^{(n)}, \tag{128}$$

$$J_{\mathrm{d}}(\underline{\mathbf{h}}_{\mathrm{T},P}^{(n)}|\underline{\mathbf{h}}_{\mathrm{S},P}^{(n)}, \underline{\mathbf{h}}_{\mathrm{F},P}^{(n-1)}) = \phi_{X_1}^2 - 2\phi_{X_1}^2 \left(\underline{\mathbf{h}}_{\mathrm{T},P}^{(n)}\right)^H \times \underline{\boldsymbol{\rho}}_{\underline{\mathbf{x}}X_1,\mathrm{SF},P}^{(n)} + \phi_{X_1}^2 \left[\left(\underline{\mathbf{h}}_{\mathrm{T},P}^{(n)}\right)^H \underline{\boldsymbol{\rho}}_{\underline{\mathbf{x}}X_1,\mathrm{SF},P}^{(n)}\right]^2, \tag{129}$$

$$J_{\mathrm{r}}(\underline{\mathbf{h}}_{\mathrm{T},P}^{(n)}|\underline{\mathbf{h}}_{\mathrm{S},P}^{(n)}, \underline{\mathbf{h}}_{\mathrm{F},P}^{(n-1)}) = \left(\underline{\mathbf{h}}_{\mathrm{T},P}^{(n)}\right)^H \mathbf{\Phi}_{\mathrm{in,SF},P}^{(n)} \underline{\mathbf{h}}_{\mathrm{T},P}^{(n)}, \tag{130}$$

$$J_{\mathrm{d}}(\underline{\mathbf{h}}_{\mathrm{F},P}^{(n)}|\underline{\mathbf{h}}_{\mathrm{S},P}^{(n)}, \underline{\mathbf{h}}_{\mathrm{T},P}^{(n)}) = \phi_{X_1}^2 - 2\phi_{X_1}^2 \left(\underline{\mathbf{h}}_{\mathrm{F},P}^{(n)}\right)^H \times \underline{\boldsymbol{\rho}}_{\underline{\mathbf{x}}X_1,\mathrm{ST},P}^{(n)} + \phi_{X_1}^2 \left[\left(\underline{\mathbf{h}}_{\mathrm{F},P}^{(n)}\right)^H \underline{\boldsymbol{\rho}}_{\underline{\mathbf{x}}X_1,\mathrm{ST},P}^{(n)}\right]^2, \tag{131}$$

$$J_{\mathrm{r}}(\underline{\mathbf{h}}_{\mathrm{F},P}^{(n)}|\underline{\mathbf{h}}_{\mathrm{S},P}^{(n)}, \underline{\mathbf{h}}_{\mathrm{T},P}^{(n)}) = \left(\underline{\mathbf{h}}_{\mathrm{F},P}^{(n)}\right)^H \mathbf{\Phi}_{\mathrm{in,ST},P}^{(n)} \underline{\mathbf{h}}_{\mathrm{F},P}^{(n)}. \tag{132}$$

By solving the constrained optimization problems (87), (88) and (89), the iterative tradeoff sub-filters are deduced, respectively, as

$$\underline{\mathbf{h}}_{\mathrm{S},P,\mu_1}^{(n)} = \phi_{X_1}^2 \left[\phi_{X_1}^2 \underline{\boldsymbol{\rho}}_{\underline{\mathbf{x}}X_1,\mathrm{TF},P}^{(n-1)} \left(\underline{\boldsymbol{\rho}}_{\underline{\mathbf{x}}X_1,\mathrm{TF},P}^{(n-1)}\right)^H + \mu_1 \mathbf{\Phi}_{\mathrm{in,TF},P}^{(n-1)}\right]^{-1} \underline{\boldsymbol{\rho}}_{\underline{\mathbf{x}}X_1,\mathrm{TF},P}^{(n-1)}, \tag{133}$$

$$\underline{\mathbf{h}}_{\mathrm{T},P,\mu_2}^{(n)} = \phi_{X_1}^2 \left[\phi_{X_1}^2 \underline{\boldsymbol{\rho}}_{\underline{\mathbf{x}}X_1,\mathrm{SF},P}^{(n)} \left(\underline{\boldsymbol{\rho}}_{\underline{\mathbf{x}}X_1,\mathrm{SF},P}^{(n)}\right)^H + \mu_2 \mathbf{\Phi}_{\mathrm{in,SF},P}^{(n)}\right]^{-1} \underline{\boldsymbol{\rho}}_{\underline{\mathbf{x}}X_1,\mathrm{SF},P}^{(n)}, \tag{134}$$

$$\underline{\mathbf{h}}_{\mathrm{F},P,\mu_3}^{(n)} = \phi_{X_1}^2 \left[\phi_{X_1}^2 \underline{\boldsymbol{\rho}}_{\underline{\mathbf{x}}X_1,\mathrm{ST},P}^{(n)} \left(\underline{\boldsymbol{\rho}}_{\underline{\mathbf{x}}X_1,\mathrm{ST},P}^{(n)}\right)^H\right.$$

$$+\mu_3 \mathbf{\Phi}^{(n)}_{\text{in,ST},P}\Big]^{-1} \underline{\boldsymbol{\rho}}^{(n)}_{\underline{\mathbf{x}}X_1,\text{ST},P}. \quad (135)$$

Applying the Woodbury's identity [63] to (133), (134) and (135), we obtain the equivalent forms of $\underline{\mathbf{h}}^{(n)}_{\text{S},P,\mu_1}$, $\underline{\mathbf{h}}^{(n)}_{\text{T},P,\mu_2}$ and $\underline{\mathbf{h}}^{(n)}_{\text{F},P,\mu_3}$, respectively, as in (94), (95) and (96).

REFERENCES

[1] J. Chen, J. Benesty, Y. Huang, and E. J. Diethorn, "Fundamentals of noise reduction," in *Springer Handbook of Speech Processing.* J. Benesty, M. M. Sondhi, and Y. Huang, Eds., pp. 843–871, Berlin, Germany: Springer-Verlag, 2008.

[2] P. C. Loizou, *Speech Enhancement: Theory and Practice*. Boca Raton, FL: CRC, 2007.

[3] J. Benesty, S. Makino, and J. Chen, *Speech Enhancement*. Berlin, Germany: Springer-Verlag, 2005.

[4] J. Benesty, J. Chen, Y. Huang, and I. Cohen, *Noise Reduction in Speech Processing*. Berlin, Germany: Springer-Verlag, 2009.

[5] Y. Ephraim and D. Malah, "Speech enhancement using a minimum mean-square error log-spectral amplitude estimator," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. ASSP-33, no. 2, pp. 443–445, Apr. 1985.

[6] S. Doclo and M. Moonen, "GSVD-based optimal filtering for single and multimicrophone speech enhancement," *IEEE Trans. Signal Process.*, vol. 50, no. 9, pp. 2230–2244, Sep. 2002.

[7] P. C. Hansen and S. H. Jensen, "Subspace-based noise reduction for speech signals via diagonal and triangular matrix decompositions: Survey and analysis," *EURASIP J. Adv. Signal Process.*, vol. 2007, pp. 1–24, Jun. 2007.

[8] X. Wang, J. Benesty, and J. Chen, "A single-channel noise cancelation filter in the short-time-fourier-transform domain," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, 2016, pp. 5235–5239.

[9] X. Wang, J. Benesty, and J. Chen, "A minimum variance partially distortionless response filter for single-channel noise reduction," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, 2017, pp. 4965–4969.

[10] J. Chen, J. Benesty, Y. Huang, and S. Doclo, "New insights into the noise reduction wiener filter," *IEEE Trans. Audio, Speech, Lang. Process.*, vol. 14, no. 4, pp. 1218–1234, Jul. 2006.

[11] J. Benesty, J. Chen, Y. Huang, and T. F. Gaensler, "Time-domain noise reduction based on an orthogonal decomposition for desired signal extraction," *J. Acoust. Soc. Amer.*, vol. 132, no. 1, pp. 452–464, 2012.

[12] S. A. Schelkunoff, "A mathematical theory of linear arrays," *Bell Syst. Tech. J.*, vol. 22, pp. 80–107, Jan. 1943.

[13] B. Rafaely, "Phase-mode versus delay-and-sum spherical microphone array processing," *IEEE Signal Process. Lett.*, vol. 12, no. 10, pp. 713–716, Oct. 2005.

[14] J. Capon, "High resolution frequency-wavenumber spectrum analysis," *Proc. IEEE*, vol. 57, no. 8, pp. 1408–1418, Aug. 1969.

[15] O. L. Frost, "An algorithm for linearly constrained adaptive array processing," *Proc. IEEE*, vol. 60, no. 8, pp. 926–935, Aug. 1972.

[16] L. J. Griffiths and C. W. Jim, "An alternative approach to linearly constrained adaptive beamforming," *IEEE Trans. Antennas Propag.*, vol. AP-30, no. 1, pp. 27–34, Jan. 1982.

[17] Y. Huang, J. Benesty, and J. Chen, *Acoustic MIMO Signal Processing: Signal and Communication Technology*. New York, NY, USA: Springer 2006.

[18] S. Gannot, D. Burshtein, and E. Weinstein, "Signal enhancement using beamforming and nonstationarity with applications to speech," *IEEE Trans. Signal Process.*, vol. 49, no. 8, pp. 1614–1626, Aug. 2001.

[19] S. Gannot and I. Cohen, "Speech enhancement based on the general transfer function GSC and postfiltering," *IEEE Speech Audio Process.*, vol. 12, no. 6, pp. 561–571, Nov. 2004.

[20] J. Chen, J. Benesty, and Y. Huang, "A minimum distortion noise reduction algorithm with multiple microphones," *IEEE Trans. Audio, Speech, Lang. Process.*, vol. 16, no. 3, pp. 481–493, Mar. 2008.

[21] J. Benesty and J. Chen, *Optimal Time-Domain Noise Reduction Filters-A Theoretical Study*. Berlin, Germany: Springer, 2011.

[22] S. Doclo, A. Spriet, J. Wouters, and M. Moonen, "Frequency-domain criterion for the speech distortion weighted multichannel wiener filter for robust noise reduction," *Speech Commun.*, vol. 49, pp. 636–656, Aug. 2007.

[23] M. Souden, J. Benesty, and S. Affes, "On the global output SNR of the parameterized frequency-domain multichannel noise reduction wiener filter," *IEEE Signal Process. Lett.*, vol. 17, no. 5, pp. 425–428, May 2010.

[24] M. Souden, J. Benesty, and S. Affes, "A study of the LCMV and MVDR noise reduction filters," *IEEE Trans. Signal Process.*, vol. 58, no. 9, pp. 4925–4935, Sep. 2010.

[25] E. A. P. Habets, J. Benesty, I. Cohen, S. Gannot, and J. Dmochowski, "New insights into the MVDR beamformer in room acoustics," *IEEE Trans. Audio, Speech, Lang. Process.*, vol. 18, no. 1, pp. 158–170, Jan. 2010.

[26] C. Pan, J. Chen, and J. Benesty, "Performance study of the MVDR beamformer as a function of the source incident angle," *IEEE Trans. Audio, Speech, Lang. Process.*, vol. 22, no. 1, pp. 67–79, Jan. 2014.

[27] S. Markovichgolan, S. Gannot, and I. Cohen, "Low-complexity addition or removal of sensors/constraints in LCMV beamformers," *IEEE Trans. Signal Process.*, vol. 60, no. 3, pp. 1205–1214, Mar. 2012.

[28] R. Talmon, I. Cohen, and S. Gannot, "Convolutive transfer function generalized sidelobe canceler," *IEEE Trans. Audio, Speech, Lang. Process.*, vol. 17, no. 7, pp. 1420–1434, Sep. 2009.

[29] J. Benesty, J. Chen, and E. Habets, *Speech Enhancement in the STFT Domain*. Berlin, Germany: Springer, 2011.

[30] Y. Huang and J. Benesty, "A multi-frame approach to the frequency-domain single-channel noise reduction problem," *IEEE Trans. Audio, Speech, Lang. Process.*, vol. 20, no. 4, pp. 1256–1269, May 2012.

[31] H. Huang, L. Zhao, J. Chen, and J. Benesty, "A minimum variance distortionless response filter based on the bifrequency spectrum for single-channel noise reduction," *Digit. Signal Process.*, vol. 33, pp. 169–179, 2014.

[32] R. Tong, G. Bao, and Z. Ye, "A higher order subspace algorithm for multichannel speech enhancement," *IEEE Signal Process. Lett.*, vol. 22, no. 11, pp. 2004-2008, Nov. 2015.

[33] R. Tong and Z. Ye, "Supplementations to the higher order subspace algorithm for suppression of spatially colored noise," *IEEE Signal Process. Lett.*, vol. 24, no. 5, pp. 668–672, May 2017.

[34] X. Jia, R. Tong, and Z. Ye, "A joint time-space-frequency filtering framework for multichannel speech enhancement via complex-valued tensor representations," *Appl. Acoust.*, vol. 145, pp. 245–254, 2019.

[35] N. Vervliet, O. Debals, L. Sorber, and L. De Lathauwer, "Breaking the curse of dimensionality using decompositions of incomplete tensors: Tensor-based scientific computing in big data analysis," *IEEE Signal Process. Mag.*, vol. 31, no. 5, pp. 71–79, Sep. 2014.

[36] A. Cichocki *et al.*, "Tensor decompositions for signal processing applications: From two-way to multiway component analysis," *IEEE Signal Process. Mag.*, vol. 32, no. 2, pp. 145–163, Mar. 2015.

[37] P. Comon, "Tensors: A brief introduction," *IEEE Signal Process. Mag.*, vol. 31, no. 3, pp. 44–53, May 2014.

[38] N. D. Sidiropoulos, L. De Lathauwer, X. Fu, K. Huang, E. E. Papalexakis, and C. Faloutsos, "Tensor decomposition for signal processing and machine learning," *IEEE Trans. Signal Process.*, vol. 65, no. 13, pp. 3551–3582, Jul. 2017.

[39] B. Barak, J. A. Kelner, and D. Steurer, "Dictionary learning and tensor decomposition via the sum-of-squares method," in *Proc. 47th Annu. ACM Symp. Theory Comput.*, 2015, pp. 143–151.

[40] M. Bousse, O. Debals, and L. De Lathauwer, "A tensor-based method for large-scale blind source separation using segmentation," *IEEE Trans. Signal Process.*, vol. 65, no. 2, pp. 346–358, Jan. 2016.

[41] M. Rupp and S. Schwarz, "A tensor LMS algorithm," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process.*, 2015, pp. 3347–3351.

[42] L. N. Ribeiro, A. L. de Almeida, and J. C. Mota, "Identification of separable systems using trilinear filtering," in *Proc. IEEE 6th Int. Workshop Comput. Adv. Multi-Sensor Adaptive Process.*, 2015, pp. 189–192.

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

[44] C. Paleologu, J. Benesty, and S. Ciochină, "Adaptive filtering for the identification of bilinear forms," *Digit. Signal Process.*, vol. 75, pp. 153–167, Apr. 2018.

[45] C. Paleologu, J. Benesty, and S. Ciochină, "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.

[46] 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/ACM Trans. Audio, Speech, Lang. Process.*, vol. 27, no. 5, pp. 903–918, May 2019.

[47] J. Benesty, I. Cohen, and J. Chen, *Array Processing-Kronecker Product Beamforming. Springer Topics in Signal Processing*, Springer-Verlag, Switzerland, 2019.

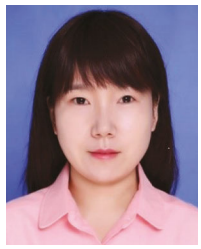
[48] M. N. da Costa, G. Favier, and J. M. T. Romano, "Tensor modelling of mimo communication systems with performance analysis and kronecker receivers," *Signal Process.*, vol. 145, pp. 304–316, 2018.
[49] L. N. Ribeiro, A. L. de Almeida, and J. C. M. Mota, "Separable linearly constrained minimum variance beamformers," *Signal Process.*, vol. 158, pp. 15–25, 2019.
[50] T. G. Kolda and B. W. Bader, "Tensor decompositions and applications," *SIAM Rev.*, vol. 51, no. 3, pp. 455–500, 2009.
[51] D. P. Bertsekas, *Nonlinear Programming*. 2nd ed. Belmont, MA, USA: Athena Scientific, 1999.
[52] A. Yener, R. D. Yates, and S. Ulukus, "Interference management for CDMA systems through power control, multiuser detection, and beamforming," *IEEE Trans. Commun.*, vol. 49, no. 7, pp. 1227–1239, Jul. 2001.
[53] T. H. Cormen, C. E. Leiserson, R. L. Rivest, and C. Stein, *Introduction to Algorithms*. Cambridge, MA, USA: MIT Press, 2009.
[54] W. C. Ward, G. W. Elko, R. A. Kubli, and W. C. McDougald, "The new varechoic chamber at AT&T bell labs," in *Proc. Wallance Clement Sabine Centennial Symp.*, 1994, pp. 343–346.
[55] J. W. Lyons, "DARPA TIMIT acoustic-phonetic continuous speech corpus," Nat. Inst. Standards Technol., Gaithersburg, MD, USA, Tech. Rep. NISTIR 4930, 1993.
[56] E. A. P. Habets, I. Cohen, and S. Gannot, "Generating nonstationary multisensor signals under a spatial coherence constraint," *J. Acoust. Soc. Amer.*, vol. 124, pp. 2911–2917, Nov. 2008.
[57] A. Varga and H. J. M. Steeneken, "Assessment for automatic speech recognition: II NOISEX-92: A database and an experiment to study the effect of additive noise on speech recognition systems," *Speech Commun.*, vol. 12, no. 3, pp. 247–51, 1993.
[58] I. Cohen, "Noise spectrum estimation in adverse environments: Improved minima controlled recursive averaging," *IEEE Trans. Speech Audio Process.*, vol. 11, no. 5, pp. 466–475, Sep. 2003.
[59] R. C. Hendriks and T. Gerkmann, "Noise correlation matrix estimation for multi-microphone speech enhancement," *IEEE Trans. Audio, Speech, Lang. Process.*, vol. 20, no. 1, pp. 223–233, Jan. 2012.
[60] ITU-T, Recommendation P.862—Perceptual evaluation of speech quality(PESQ): An objective method for end-to-end speech quality assessment of narrow-band telephone networks and speech codecs, International Tele-communication Union-Telecommunication Standardisation Sector (ITU-T), Feb. 2001
[61] 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.
[62] ITU-T, Recommendation P.862.1—Mapping function for transforming P.862raw result scores to MOS-LQO, International Telecommunication Union-Telecommunication Standardisation Sector (ITU-T), Mar. 2005.
[63] J. N. Franklin, *Matrix Theory*. Englewood Cliffs, NJ, USA: Prentice-Hall, 1968.

Xianghui Wang received the bachelor's degree in electronics and information engineering from Xi'an University, Xi'an, China, in 2008, and the master's degree in signal and information processing and the Ph.D. degree in information and communication engineering from Northwestern Polytechnical University, Xi'an, China, in 2011 and 2020, respectively. During 2016–2018, he was also a visiting Ph.D. student with the the Andrew and Erna Viterby Faculty of Electrical Engineering, Technion – Israel Institute of Technology, Haifa, Israel. He is currently a Lecturer with the School of Electronic Information and Artificial Intelligence, Shaanxi University of Science and Technology, Xi'an, China. His research interests include speech enhancement and microphone array signal processing.

Jie Chen (Senior Member, IEEE) received the B.S. degree from Xi'an Jiaotong University, Xi'an, China, in 2006, the Dipl.-Ing. degree in information and telecommunication engineering from the University of Technology of Troyes (UTT), Troyes, France, in 2009, the M.S. degree in information and telecommunication engineering from Xi'an Jiaotong University, in 2009, and the Ph.D. degree in systems optimization and security from UTT, in 2013.

From 2013 to 2014, he was with the Lagrange Laboratory, University of Nice Sophia Antipolis, Nice, France. From 2014 to 2015, he was with the Department of Electrical Engineering and Computer Science, University of Michigan, Ann Arbor, MI, USA. He is currently a Professor with Northwestern Polytechnical University, Xi'an, China. His research interests include adaptive signal processing, distributed optimization, hyperspectral image analysis, and acoustic signal processing.

He was the Technical Co-Chair of IWAENC'16 held in Xi'an, China. He is a Distinguished Lecturer of Asia-Pacific Signal and Information Processing Association (2018C2019), the Co-Chair of IEEE Signal Processing Society Summer School 2019 in Xi'an, and the Technical Co-Chair of IEEE ICSPCC 2021. He will be the General Co-Chair of IEEE MLSP 2022.



Xiaoyi Chen was born in 1986. She received the bachelor's degree in communication engineering from the Xi'an University of Posts and Telecommunications, Xi'an, China, in 2008, and the master's and Doctoral degrees in underwater acoustic engineering from Northwestern Polytechnical University, Xi'an, China, in 2011 and 2016, respectively. She is currently a Lecturer with the Shaanxi University of Science and Technology, Xi'an, China. Her research interests include speech separation and microphone array signal processing.

Jing Guo received the Ph.D. degree in communication and signal processing from the State Key Laboratory of Integrated Services Networks, Xidian University, Xi'an, China. She is currently a Lecturer with the School of Electronic Information and Artificial Intelligence, Shaanxi University of Science and Technology, Xi'an, China. Her current research interests include NOMA cooperative systems, reconfigurable intelligent surface (RIS), and wireless communication theory.

Qian Xiang was born in1986. He received the bachelor's degree in electronics and information engineering from Anhui Xinhua College, China, in 2010 and the master's degree in signal and information processing from the Jiangxi Science and Technology Normal University, Nanchang, China, in 2013. He is currently working toward the Ph.D. degree in process system engineering with the Shaanxi University of Science and Technology, Xi'an, China. From 2015 to 2020, he was with Fuyang Normal University, Fuyang, China. His research interests include speech enhancement and microphone array signal processing.