A Unified Statistical Approach to Fast and Robust Multichannel Speech Separation and Dereverberation

Kouhei Sekiguchi

Abstract

This thesis describes a unified statistical approach to joint multichannel source separation and dereverberation. This technique is useful as a front end of various audio applications including smart speakers, conversational robots, and hearing aid systems because recorded signals usually consist of utterances of target and non-target speakers that often overlap, environmental noise, and reverberation. The objective of source (speech) separation is to recover multiple source (speech) signals from observed multichannel mixture signals. Speech enhancement is one important type of speech separation that aims to extract only utterances of a particular speaker from noisy mixture signals. In addition, it is necessary to recover anechoic (dry) speech signals for improving the speech intelligibility and the performance of automatic speech recognition.

A typical approach to multichannel source separation is to formulate and optimize a unified probabilistic model consisting of a *source model* representing the power spectral densities (PSDs) of sources and a *spatial model* representing their spatial covariance matrices (SCMs). Assuming that the PSDs of all sources have low-rank structures in the time-frequency domain, nonnegative matrix factorization (NMF) has often been used for formulating a source model. One of the most successful examples of this approach is multichannel nonnegative matrix factorization (MNMF) consisting of a *low-rank source model* based on NMF and a *full-rank spatial model* assuming the full-rankness of the SCMs for dealing with reverberation longer than a window size. Although MNMF is a versatile blind source separation method that has a tuning-free convergence-guaranteed iterative optimization algorithm, it has four major problems. 1) The low-rankness of the source PSDs does not always hold in reality, especially for speech sources

having complicated dynamics. 2) MNMF tends to easily get stuck in a local optimum because of a high degree of freedom of the spatial model. 3) The optimization algorithm is too computationally expensive. 4) The performance of MNMF is severely degraded under a realistic echoic condition.

To solve the problems 1) and 2), in Chapter 3, we propose a semi-blind speech enhancement method called MNMF-DSP that uses a conventional low-rank model and a deep generative model as noise and speech models (source models), respectively. While the noise model is learned *on the fly* from observed noisy speech signals in an unsupervised manner, the speech model is learned *in advance* from clean speech signals in an unsupervised manner and used as a prior of clean speech spectrogram. We experimentally show that MNMF-DSP outperformed MNMF and alleviates the initialization sensitivity.

To solve the problems 2) and 3), in Chapter 4, we propose a computationally-efficient variant of MNMF called FastMNMF based on a jointly-diagonalizable full-rank spatial model. Assuming the SCMs of all sources to be jointly diagonalizable, computationally-expensive MNMF dealing with the inter-channel covariance can be converted to light-weight nonnegative tensor factorization (NTF) based on the inter-channel independence. To explicitly consider the directivity or diffuseness of each source, we also propose rank-constrained FastMNMF that enables us to individually specify the ranks of SCMs. We experimentally show the superiority of FastMNMF over MNMF and the effectiveness of the rank constraint.

To solve the problem 4), in Chapter 5, we propose an extension of FastMNMF based on an autoregressive-moving average (ARMA) model called ARMA-FastMNMF for joint blind source separation and dereverberation. The early part of the reverberation is represented by the MA model, and the late part is mainly represented by the AR model, which is suitable for representing long reverberations. To derive efficient update rules, we introduce the joint-diagonalization constraint on the MA model. We experimentally show that ARMA-FastMNMF outperforms conventional methods in many situations.

In Chapter 6, we conclude this thesis with a brief look at future work for real-time joint separation and dereverberation of a varying number of sources.

Acknowledgment

This work was accomplished at Speech and Audio Processing Laboratory, Graduate School of Informatics, Kyoto University. I express my gratitude to all people who helped me and this work.

At first, I would like to express my special thanks and appreciation to my supervisor Associate Professor Kazuyoshi Yoshii. He gave me the opportunity to learn in Speech and Audio Processing Lab. His comments were essential and insightful for advancing this work. This work would not have been completed without his continuing engagement and generous support.

I also express my special thanks and appreciation to Professor Tatsuya Kawahara for a lot of insightful comments on my research.

Furthermore, I express my special thanks and appreciation to the members of my dissertation committee, Professor Toshiyuki Tanaka and Professor Ko Nishino for their time, valuable comments, and suggestions.

This thesis cannot be accomplished without Dr. Yoshiaki Bando, the researcher at National Institute of Advanced Industrial Science and Technology. He supported me and gave much time to meaningful discussions for a long time.

I wish to deeply thank Dr. Aditya Arie Nugraha and Dr. Mathieu Fontaine, researchers at RIKEN AIP. I received a great deal of inspiration from discussions with these members.

I also express my thanks to the members in Speech and Audio Processing Lab. I am grateful to comments and supports from Dr. Eita Nakamura, Dr. Shinsuke Sakai, Mr. Masato Mimura, Dr. Ryo Nishikimi, and the other members.

Lastly, I sincerely thank my family for their support and encouragement for my long student life.

Contents

Chapter 1

Introduction

1.1 Background

Audio signal processing has been used in various applications such as smart speakers (*e.g.*, Google Home and Amazon Alexa), conversational robots, audio scene analysis, and hearing aid systems [1–5]. In such applications, the recorded signals are often contaminated with utterances of non-target speakers, environmental noise, and reverberation because the target sound source is not always close to the microphones of the systems. In smart speakers, conversational robots, or hearing aid systems, the system typically needs to extract one target utterance from noisy audio recording, *i.e.*, computationally realize the cocktail-party effect. This task is called *speech enhancement* [6–14]. Since the reverberation makes the speech unclear and degrades the performance of automatic speech recognition (ASR), *speech dereverberation* is also necessary [15–20]. In conversational robots or audio scene analysis, the system often needs to extract individual sound sources from the mixture signals because, for example, it is necessary to deal with the overlap of multiple utterances (*e.g.*, barge-ins). This task is called *speech separation* [21–30]. To handle such tasks, multichannel signal processing is widely used. The advantage of using multichannel signals is the availability of both the phase and level differences of the observed signals between microphones.

1.2 Requirements

There are three main technical requirements in multichannel speech enhancement, separation, and/or dereverberation.

Blindness: To use a signal processing system in a wide range of acoustic environments, it should be free from prior information about recording environments and sound sources. In recent years, supervised methods using deep neural networks (DNNs) have been actively studied [9, 10, 31, 32]. Such methods require paired data of noisy and clean speech signals for speech enhancement, paired data of mixture and source signals for source separation, and paired data of reverberant and anechoic signals for dereverberation. Because the statistical characteristics of observed mixtures significantly vary according to the length of reverberation and the types of sound sources, it is impossible to cover all possible conditions. Since such supervised methods tend to be unstable for unseen data, they may be suitable only for a system used in a specific situation [33, 34]. Thus, we focus on the blind or semi-blind approach in this thesis.

Computational efficiency: Applications such as conversational robots and smart speakers [3–5] need to recognize human speech in real time for natural quick response. Therefore, source separation and dereverberation should be computationally efficient.

Total optimality: Since the real recordings often include non-target speech and environmental noise in addition to reverberation, both speech dereverberation and separation (enhancement) are required. One naive approach to improving the speech intelligibility and the performance of ASR is to sequentially perform dereverberation and source separation (in the reverse order) for reverberant noisy recorded signals. This approach, however, is sub-optimal because the dereverberation and separation processes have mutually-dependent relationships. This calls for joint source separation and dereverberation.

1.3 Formulation

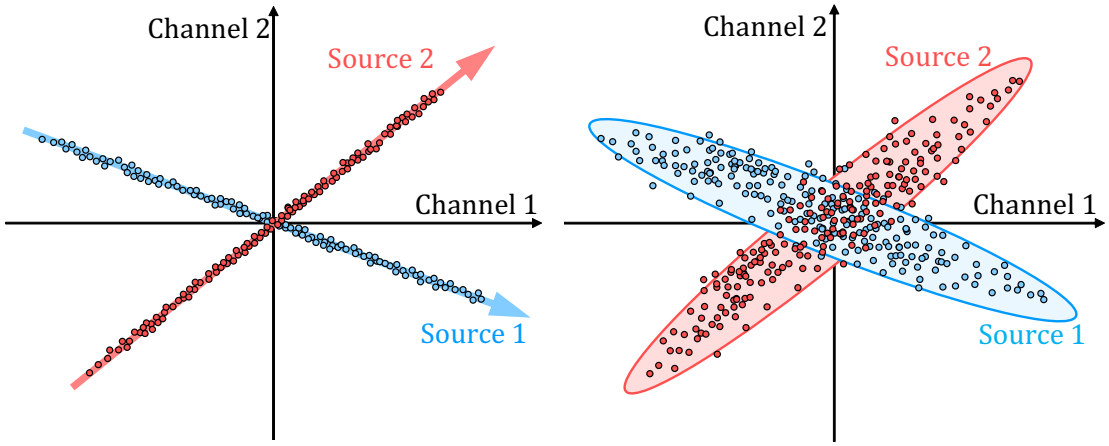
We define the general form of the source separation problem, which is referred to through this thesis. Suppose that a mixture of N sources are recorded by M microphones. Let $\mathbf{X} = \{\mathbf{x}_{ft}\}_{f=1,t=1}^{F,T} \in \mathbb{C}^{F\times T\times M}$ be the observed multichannel mixture spectrograms, where F and T represent the number of frequency bins and that of time frames, respectively. Let $\mathbf{S}_n = \{s_{nft}\}_{f=1,t=1}^{F,T} \in \mathbb{C}^{F\times T}$ be the single-channel spectrogram of source n and $\mathbf{X}_n = \{\mathbf{x}_{nft}\}_{f=1,t=1}^{F,T} \in \mathbb{C}^{F\times T\times M}$ be the multichannel spectrogram of source n called *image*, which is the contribution of source n to the observed multichannel mixture spectrogram. Assuming the additivity of the complex source spectra $\{\mathbf{x}_{nft}\}_{n=1}^{N}$, the complex mixture spectrum $\mathbf{x}_{ft} \in \mathbb{C}^M$ is given by

$$\mathbf{x}_{ft} = \sum_{n=1}^{N} \mathbf{x}_{nft}. \tag{1.1}$$

This thesis mainly focuses on blind source separation (BSS) based on unsupervised learning of a probabilistic model that represents a multichannel mixture spectrogram as the sum of multichannel source images [24–28, 35–37]. Such a probabilistic model typically consists of a *source model* representing the time-frequency (TF) structure of source spectrograms (Section 1.3.1) and a *spatial model* representing their inter-channel covariance structure (Section 1.3.2). In particular, the low-rank source model based on nonnegative matrix factorization (NMF) [8, 38] has widely been used for mitigating the permutation problem, *i.e.*, source component alignment over all frequency bins. In a typical spatial model, the TF bins of each source image are assumed to independently follow multivariate complex Gaussian distributions with full-rank or rank-1 spatial covariance matrices (SCMs).

1.3.1 Source Modeling

We formulate a source model that represents the generative process of the complex spectrum s_{nft} of each source n. s_{nft} is assumed to independently follows



(a) Rank-1 model (b) Full-rank model

Figure 1.1: Two variants of spatial models. Blue and red dots indicate source images $\{\mathbf{x}_{ft1}\}_{t=1}^{T}$ and $\{\mathbf{x}_{ft2}\}_{t=1}^{T}$ in frequency f, respectively. In the rank-1 model, dots are distributed on steering vectors $\mathbf{a}_{1f}$ and $\mathbf{a}_{2f}$. In the full-rank model, dots are widely and elliptically distributed.

circularly-symmetric complex Gaussian distribution:

$$s_{nft} \sim \mathcal{N}_{\mathbb{C}}(0, \lambda_{nft}), \tag{1.2}$$

where $\mathcal{N}_{\mathbb{C}}(\mu, \sigma^2)$ indicates a univariate circularly-symmetric complex Gaussian distribution with mean μ and variance σ^2, and λ_{nft} indicates the power spectral density (PSD) of source n at frequency f and time t.

1.3.2 Spatial Modeling

We formulate a spatial model that represents the sound propagation process between each source n and the M microphones. Two variants of spatial models (Fig. 1.1), a full-rank model with full-rank SCMs and a rank-1 model with rank-1 SCMs, have been widely used [24, 25, 27, 28]. In the rank-1 model, we assume a time-invariant linear mixing system as follows:

$$\mathbf{x}_{nft} = \mathbf{a}_{nf} s_{nft}, \tag{1.3}$$

where $\mathbf{a}_{nf} \in \mathbb{C}^M$ is the steering vector of source n at frequency f. Substituting Eq. (1.3) into Eq. (1.1), we get

$$\mathbf{x}_{ft} = \mathbf{A}_f \mathbf{s}_{ft}, \tag{1.4}$$

where $\mathbf{A}_f \triangleq [\mathbf{a}_{1f}, \ldots, \mathbf{a}_{Nf}] \in \mathbb{C}^{M \times N}$ is called a mixing matrix. Using Eqs. (1.2) and (1.3), we say

$$\mathbf{x}_{nft} \sim \mathcal{N}_{\mathbb{C}} \left(\mathbf{0}_M, \lambda_{nft} \mathbf{G}_{nf} \right), \tag{1.5}$$

where $\mathbf{G}_{nf} = \mathbf{a}_{nf} \mathbf{a}_{nf}^{\mathsf{H}} \in \mathbb{S}_+^M$ is the rank-1 SCM of source n at frequency f and $\mathbb{S}_+^M$ indicates the set of positive semi-definite matrices of size M. Using Eqs. (1.1) and (1.5), and the reproductive property of the Gaussian distribution, we say

$$\mathbf{x}_{ft} \sim \mathcal{N}_{\mathbb{C}} \left(\mathbf{0}_M, \sum_{n=1}^{N} \lambda_{nft} \mathbf{G}_{nf} \right). \tag{1.6}$$

To make the covariance matrix full-rank, the rank-1 model is applicable only in a determined ($N = M$) or underdetermined ($N > M$) condition.

As shown in Eq. (1.5), $\mathbf{G}_{nf}$ is a rank-1 matrix in an idealized situation. However, in a real indoor environment, $\mathbf{G}_{nf}$ can be a full-rank matrix due to reverberation and reflection. Therefore, in the full-rank spatial model, we assume $\mathbf{G}_{nf}$ is a full-rank matrix. The number of parameters of a full-rank SCM is $M(M+1)/2$ and that of a rank-1 SCM is only M. While the rank-1 model is a restricted version of the full-rank model, independent low-rank matrix analysis (ILRMA) based on the rank-1 model [28] is empirically known to work better than MNMF based on the full-rank model [27] because the rank-1 model is less sensitive to parameter initialization.

1.3.3 Source Separation

If the parameters of the probabilistic generative model are given, we can perform statistical source separation.

Full-Rank Model

To estimate the source image $\mathbf{x}_{nft} \in \mathbb{C}^M$, we use a multichannel Wiener filter (MWF). Using Eq. (1.5) and Eq. (1.6), the posterior expectation of the speech image $\hat{\mathbf{x}}_{nft} \in \mathbb{C}^M$ is given as follows:

$$\hat{\mathbf{x}}_{nft} = \mathbb{E}[\mathbf{x}_{nft} | \mathbf{x}_{ft}] = \mathbf{Y}_{nft} \mathbf{Y}_{ft}^{-1} \mathbf{x}_{ft}, \tag{1.7}$$

where $\mathbf{Y}_{nft} \triangleq \lambda_{nft} \mathbf{G}_{nf}$ and $\mathbf{Y}_{ft} \triangleq \sum_{n=1}^{N} \mathbf{Y}_{nft}$.

Rank-1 Model

To estimate the source image $s_{nft} \in \mathbb{C}$, we use a linear demixing filter as follows:

$$\hat{s}_{nft} = \mathbf{d}_{nf}^{\mathsf{H}} \mathbf{x}_{ft}. \tag{1.8}$$

To solve the scale ambiguity of $\{\hat{s}_{nft}\}_{f=1}^{F}$ over frequency bins, we use a projection back technique [39] for estimating the source image $\hat{\mathbf{x}}_{nft} \in \mathbb{C}^M$ as follows:

$$\hat{\mathbf{x}}_{nft} = \mathbf{a}_{nf} \hat{s}_{nft} = \mathbf{a}_{nf} \mathbf{d}_{nf}^{\mathsf{H}} \mathbf{x}_{ft}. \tag{1.9}$$

When $\mathbf{G}_{nf}$ is a rank-1 matrix given by $\mathbf{G}_{nf} = \mathbf{a}_{nf}\mathbf{a}_{nf}^{\mathsf{H}}$, $\mathbf{Y}_{ft}$ is given as follows:

$$\begin{aligned} \mathbf{Y}_{ft} &= \sum_{n=1}^{N} \lambda_{nft} \mathbf{a}_{nf} \mathbf{a}_{nf}^{\mathsf{H}} \\ &= \mathbf{A}_f \mathbf{\Lambda}_{ft} \mathbf{A}_f^{\mathsf{H}} \\ &= \mathbf{D}_f^{-1} \mathbf{\Lambda}_{ft} \mathbf{D}_f^{-\mathsf{H}}, \end{aligned} \tag{1.10}$$

where $\mathbf{\Lambda}_{ft} \triangleq \mathrm{Diag}(\lambda_{1ft}, \ldots, \lambda_{Nft})$ and $\mathbf{D}_f \triangleq \mathbf{A}_f^{-1}$ is called a demixing matrix. Substituting Eq. (1.10) into Eq. (1.7), we can easily prove that Eq. (1.9) can also be obtained by the MWF as follows:

$$\hat{\mathbf{x}}_{nft} = \left(\lambda_{nft} \mathbf{a}_{nf} \mathbf{a}_{nf}^{\mathsf{H}}\right) \left(\mathbf{D}_f^{\mathsf{H}} \mathbf{\Lambda}_{ft}^{-1} \mathbf{D}_f\right) \mathbf{x}_{ft} \tag{1.11}$$

$$= \lambda_{nft} \mathbf{a}_{nf} \mathbf{e}_n^{\mathsf{T}} \mathbf{\Lambda}_{ft}^{-1} \mathbf{D}_f \mathbf{x}_{ft} \tag{1.12}$$

$$= \mathbf{a}_{nf} \mathbf{d}_{nf}^{\mathsf{H}} \mathbf{x}_{ft}, \tag{1.13}$$

where $\mathbf{e}_n$ is a one-hot vector whose n-th element is one.

1.4 Approach

Multichannel nonnegative matrix factorization (MNMF) [25–27] and its special case called independent low-rank matrix analysis (ILRMA) [28] are representative BSS methods. MNMF consists of a low-rank source model and a full-rank spatial model, and ILRMA consists of a low-rank source model and a rank-1 spatial model. Although MNMF is a versatile blind source separation method that has a tuning-free convergence-guaranteed iterative optimization algorithm, it has four

major problems. 1) The low-rankness of the source PSDs does not always hold in reality, especially for speech sources having complicated dynamics. 2) MNMF tends to easily get stuck in a local optimum because of a high degree of freedom of the spatial model. 3) The optimization algorithm is too computationally expensive. 4) The performance of MNMF is severely degraded under a realistic echoic condition.

To solve the problems 1) and 2), in Chapter 3, we propose a semi-blind speech enhancement method called MNMF-DSP that uses a conventional low-rank model and a deep generative model as noise and speech models (source models), respectively. The low-rank source model is suitable for only particular types of audio signals such as music and stationary noise and cannot represent complex structures of speech signals, resulting in low separation performance. We thus use the deep generative model of speech (called a deep speech prior in speech enhancement) trained from only clean speech signals instead of the low-rank speech model [40, 41]. While the noise model is learned *on the fly* from observed noisy speech signals in an unsupervised manner, the speech model is learned *in advance* from clean speech signals in an unsupervised manner and used as a prior of clean speech spectrogram. Thus, this approach can work in various acoustic environments. While the low-rank source model could loosely fit any types of source spectrograms within its representation capability, the deep generative model of speech can precisely represent only speech spectrograms (*e.g.*, harmonic structures). This leads to the excellent performance of speech enhancement and the stability of parameter estimation, compared to blind methods.

To solve the problems 2) and 3), in Chapter 4, we propose a computationally-efficient variant of MNMF called FastMNMF based on a jointly-diagonalizable (JD) full-rank spatial model. To reduce the model complexity, ILMRA restricts the SCMs to rank-1 matrices [28]. Its performance, however, is limited because the rank-1 constraint does not hold in a real echoic environment with diffuse noise [42,43]. The basic version of FastMNMF (called FastMNMF1) [44,45] instead restricts the SCMs of all sources to JD yet full-rank matrices in a frequency-wise manner. We derive a convergence guaranteed update rule for *diagonalizers*, which

jointly diagonalize the SCMs, using an iterative projection (IP) algorithm, while a fixed point iteration (FPI) algorithm without convergence guarantee is used in [44]. Considering the interpretation of the JD full-rank spatial model, we propose a constrained version of FastMNMF1 (called FastMNMF2) that shares the direction weights of each source over all frequency bins. To explicitly consider the directivity or diffuseness of each source, we also propose rank-constrained FastMNMF that enables us to individually specify the ranks of SCMs. We experimentally show FastMNMF worked better than MNMF and ILRMA, and is an order of magnitude faster than MNMF.

To solve the problem 4), in Chapter 5, we propose an extension of FastMNMF2 based on an autoregressive-moving average (ARMA) model called ARMA-FastMNMF2 for joint blind source separation and dereverberation. Reverberations are often represented by a moving average (MA) model [15, 17]. When the reverberation time is long, however, the tap length of the MA model becomes long, that is, the number of parameters becomes quite large. To alleviate this problem, an autoregressive (AR) model [18, 19] is introduced to represent the late part of the reverberation (late reverberation), and the MA model is used to represent the early part (early reflection), resulting in the ARMA model [46]. The AR model can represent infinitely-long reverberations with a finite tap length in theory. If it is used for representing the early reflections, however, it may also represent the direct speech signals because of the correlations inherent in the speech signals. To derive an efficient update rules for the AR coefficients, we introduce the joint-diagonalization constraint on the MA model. Because the MA model may also represent a part of the direct signals, we further introduce the rank-constraint to the SCMs of the MA model to keep them away from those of the direct signals. We experimentally show that ARMA-FastMNMF2 outperforms AR- or MA-based extensions of FastMNMF2. in many situations.

The organization of this thesis is outlined in Fig. 1.2. Chapter 2 provides a literature review about conventional blind and supervised source separation methods and dereverberation methods. Chapter 3 presents a semi-blind speech enhancement method with a deep speech prior. Chapter 4 presents a fast blind

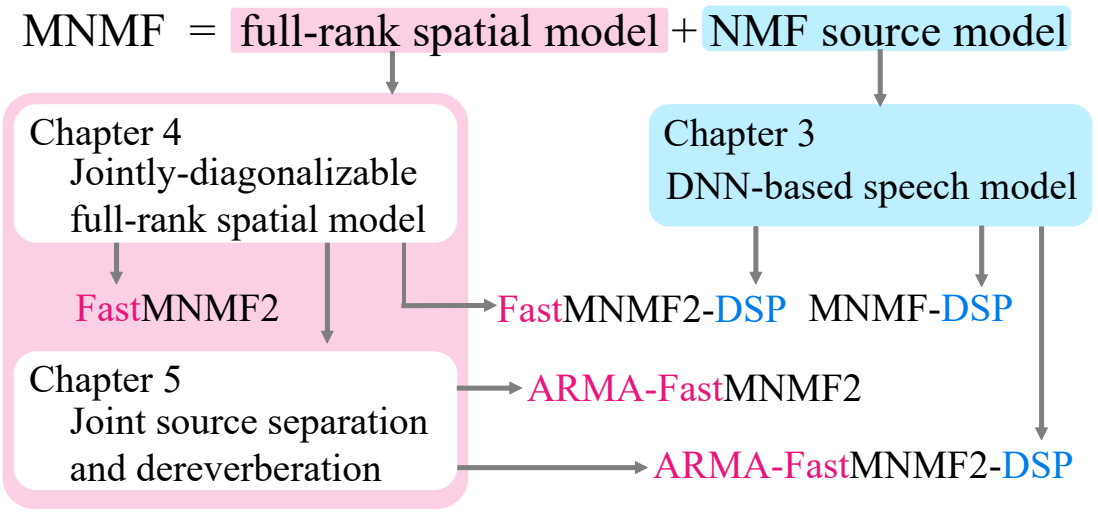


Figure 1.2: Organization of this thesis.

source separation method with a jointly-diagonalizable full-rank spatial model. Chapter 5 presents a fast joint blind source separation and dereverberation method with a deep speech prior. Finally, Chapter 6 concludes this thesis.

Chapter 2

Literature Review

This chapter reviews the literature related to sound source separation, speech enhancement, and dereverberation. Section 2.1 summarizes the existing sound source separation and speech enhancement methods. Section 2.2 summarizes the existing dereverberation methods.

2.1 Source Separation and Speech Enhancement

We categorize source separation and speech enhancement methods into single-channel and multichannel methods. For single-channel case, we mainly focus on the methods based on nonnegative matrix factorization (NMF) and deep learning. Multichannel methods are categorized into unsupervised blind methods and the methods based on deep learning.

2.1.1 Single-channel Methods Based on Nonnegative Matrix Factorization

Nonnegative matrix factorization (NMF) [38] has widely been used for unsupervised or supervised source separation. It approximates a nonnegative matrix $\mathbf{X} \triangleq [\mathbf{x}_1, \cdots, \mathbf{x}_T] \in \mathbb{R}^{F \times T}$ as the product of a basis matrix $\mathbf{W} \triangleq [\mathbf{w}_1, \cdots, \mathbf{w}_K] \in \mathbb{R}_+^{F \times K}$ and an activation matrix $\mathbf{H} \triangleq [\mathbf{h}_1, \cdots, \mathbf{h}_K]^\mathsf{T} \in \mathbb{R}_+^{K \times T}$, *i.e.*, $\mathbf{X}$ is approximated by the sum of K rank-1 matrices $\{\mathbf{w}_k \mathbf{h}_k^\mathsf{T}\}_{k=1}^K$. Thanks to the nonnegative and additive natures of NMF, the basis vectors $\{\mathbf{w}_k\}_{k=1}^K$ tend to represent repeatedly-used *parts*

such that $\mathbf{x}_t$ is efficiently represented by a weighted combination of $\{\mathbf{w}_k\}_{k=1}^K$, *i.e.*,

$$x_{tf} \approx \sum_{k=1}^{K} w_{kf} h_{kt}. \tag{2.1}$$

The parameters $\mathbf{W}$ and $\mathbf{H}$ are estimated such that the approximation error between $\mathbf{X}$ and $\mathbf{WH}$ is minimized. The β-divergences with $\beta = 0, 1, 2$ corresponding to the Itakura-Saito (IS) and Kullback-Leibler (KL) divergences and Euclidean distance, respectively, have often been used [47].

When NMF is used for single-channel source separation, the power spectrogram of an observed mixture signal is given as a nonnegative matrix $\mathbf{X}$ and the estimated $\mathbf{W}$ and $\mathbf{H}$ represent a set of *time-invariant* basis spectra and a set of the corresponding *time-varying* activations, respectively, where each "basis" does not always correspond to a "source." Because the power spectrogram of a realistic source signal is not a rank-1 matrix in general, one may cluster K rank-1 matrices $\{\mathbf{w}_k\mathbf{h}_k^\mathsf{T}\}_{k=1}^K$ into N sources in a post-processing step. Given the source types included in the mixture signal, another solution is to train bases for each source beforehand and estimate only activations for the observed mixture. If the source types of some of the sources are known, only the bases for the known sources can be trained beforehand.

NMF with the IS divergence (IS-NMF) [8] is often used as a theoretically-reasonable choice because it is equivalent to the maximum likelihood estimation of a probabilistic generative model of the mixture signal. Assuming the additivity of complex spectrograms, the short-time Fourier transform (STFT) coefficient s_{ft} of the mixture spectrogram is given by $s_{ft} = \sum_{k=1}^K s_{kft}$ at frequency f and time t, where each basis spectrogram s_{kft} is assumed to follow a circularly-symmetric complex Gaussian distribution as follows:

$$s_{kft} \sim \mathcal{N}_\mathbb{C}\left(0, w_{kf} h_{kt}\right). \tag{2.2}$$

Using the reproductive property of the Gaussian distribution, s_{ft} can be said to follow a circularly-symmetric complex Gaussian distribution as follows:

$$s_{ft} \sim \mathcal{N}_\mathbb{C}\left(0, \sum_{k=1}^{K} w_{kf} h_{kt}\right). \tag{2.3}$$

The maximization of this log-likelihood is equivalent to the minimization of the IS-divergence between $\mathbf{S} \triangleq \{|s_{ft}|^2\}_{f,t=1}^{F,T}$ and $\mathbf{WH}$. Note that in practice, $x_{tf} = |s_{ft}|^2$ is often assumed to follow a Poisson distribution, resulting in KL-NMF [48], because KL-NMF is experimentally known to work better than IS-NMF even when $\mathbf{W}$ and $\mathbf{H}$ are randomly initialized. Given s_{ft}, each basis s_{kft} is inferred with single-channel Wiener filtering as follows:

$$\mathbb{E}[s_{kft} \mid s_{ft}] = \frac{w_{kf}h_{kt}}{\sum_{k'=1}^{K} w_{k'f}h_{k't}} s_{ft}. \tag{2.4}$$

Correlated tensor factorization (CTF) [49] is an ultimate extension of NMF that approximates a positive semidefinite matrix $\tilde{\mathbf{X}} \in \mathbb{S}_+^{FT}$ as the sum of the Kronecker products of two sets of positive semidefinite matrices, $\mathbf{V}_k \in \mathbb{S}_+^F$ and $\mathbf{U}_k \in \mathbb{S}_+^T$, as follows:

$$\tilde{\mathbf{X}} \approx \sum_{k=1}^{K} \mathbf{V}_k \otimes \mathbf{U}_k, \tag{2.5}$$

where $\otimes$ represents the Kronecker product. CTF with the log-determinant divergence (LD-CTF) can be used for audio source separation considering the frequency and time covariance structures, whereas NMF assumes that all time-frequency bins are independent. Let $\mathbf{s}_k \in \mathbb{C}^{FT}$ and $\mathbf{s} = \sum_{k=1}^{K} \mathbf{s}_k$ be the complex vectors obtained by serializing the source and mixture complex spectrograms, respectively. $\mathbf{s}_k$ is assumed to follow a multivariate complex Gaussian distribution as follows:

$$\mathbf{s}_k \sim \mathcal{N}_{\mathbb{C}}\left(\mathbf{0}_{FT}, \mathbf{V}_k \otimes \mathbf{U}_k\right), \tag{2.6}$$

Using Eq. (2.6) and the additive property of the Gaussian distribution, $\mathbf{s}$ also follows a circularly-symmetric complex Gaussian distribution as follows:

$$\mathbf{s} \sim \mathcal{N}_{\mathbb{C}}\left(\mathbf{0}_{FT}, \sum_{k=1}^{K} \mathbf{V}_k \otimes \mathbf{U}_k\right). \tag{2.7}$$

The parameters $\mathbf{V}_k$ and $\mathbf{U}_k$ are estimated in a maximum likelihood manner, and it is equivalent to the minimization of the LD divergence between $\tilde{\mathbf{S}} \triangleq \mathbf{s}\mathbf{s}^{\mathsf{H}}$ and $\sum_{k=1}^{K} \mathbf{V}_k \otimes \mathbf{U}_k$. When either $\{\mathbf{V}_k\}_{k=1}^{k}$ or $\{\mathbf{U}_k\}_{k=1}^{k}$ are diagonal matrices, LD-CTF

reduces to positive semidefinite tensor factorization based on the log-determinant divergence (LD-PSDTF) [50], and when both $\{\mathbf{V}_k\}_{k=1}^k$ and $\{\mathbf{U}_k\}_{k=1}^k$ are diagonal matrices, LD-CTF reduces to IS-NMF. One problem of CTF is its extremely high computational cost due to the huge covariance matrix of size FT.

To reduce the computational cost of CTF to a manageable level, independent low-rank tensor analysis (ILRTA called FastCTF later) [51] restricts the frequency and time covariance matrices $\{\mathbf{V}_k\}_{k=1}^k$ and $\{\mathbf{U}_k\}_{k=1}^k$ to jointly-diagonalizable matrices as follows.

$$\forall k, \quad \mathbf{V}_k = \mathbf{P}^{-1}\mathrm{Diag}(\mathbf{v}_k)\mathbf{P}^{-\mathsf{H}}, \tag{2.8}$$

$$\forall k, \quad \mathbf{U}_k = \mathbf{R}^{-1}\mathrm{Diag}(\mathbf{u}_k)\mathbf{R}^{-\mathsf{H}}, \tag{2.9}$$

where $\mathbf{v}_k \in \mathbb{R}_+^F$ and $\mathbf{u}_k \in \mathbb{R}_+^T$ are nonnegative vectors and $\mathbf{P} \in \mathbb{S}_+^F$ and $\mathbf{R} \in \mathbb{S}_+^T$ are non-singular matrices called *diagonalizers*. This means that a spectrogram-like representation $\mathbf{P}\mathbf{X}\mathbf{R}^{\mathsf{T}} \in \mathbb{C}^{F\times T}$ obtained by linear projection of $\mathbf{X}$ with $\mathbf{P}$ and $\mathbf{R}$ follows a multivariate complex Gaussian distribution with a diagonal covariance matrix given by $\sum_{k=1}^K \mathrm{Diag}(\mathbf{v}_k) \otimes \mathrm{Diag}(\mathbf{u}_k)$. Because each TF bin of $\mathbf{P}\mathbf{X}\mathbf{R}^{\mathsf{T}}$ independently follows a univariate complex Gaussian distribution, CTF for $\mathbf{X}$ is equivalent to computationally-efficient IS-NMF for $\mathbf{P}\mathbf{X}\mathbf{R}^{\mathsf{T}}$. The same idea is used for restricting the spatial covariance matrices (SCMs) in Chapter 4.

2.1.2 Single-channel Methods Based on Deep Learning

Deep neural networks (DNNs) have intensively been used for supervised non-blind source separation and speech enhancement in the time-frequency (TF) domain [9, 10, 31, 52–60] or in the time domain [61–63]. A typical approach is to train a DNN that estimates TF masks [10, 31, 53] by using paired data of mixture and source spectrograms. The complex source spectrograms are obtained by dividing the complex mixture spectrogram according to the estimated masks. Several types of masks have been proposed, *e.g.*, the ideal binary mask (IBM) [64], which takes one if the signal-to-noise ratio (SNR) of a TF bin exceeds a threshold and takes zero otherwise, the ideal ratio mask (IRM) [10], which is the ratio of the target power and the mixture power, and the phase sensitive mask (PSM) [31],

which is the product of the IRM and the cosine of the phase difference between the target and recorded signals.

Instead of evaluating the estimated TF masks, magnitude spectrum approximation (MSA) [9, 52] evaluates the estimated source spectrograms (obtained after TF masking) in the training phase. In speech enhancement, the denoising autoencoder (DAE) [9] has often been used for directly converting a noisy speech spectrogram to a clean speech spectrogram. In speech separation, the estimated speech sources should be associated with the reference sources for supervised training. A popular strategy is to use permutation invariant training (PIT) [56, 57], in which the cost functions for all possible combinations are calculated and then the lowest value is used for calculating gradient.

As another approach to the permutation problem, one may use the deep attractor network (DAN) [58]. It computes the embeddings of the TF bins of a mixture spectrogram and then computes their source-wise averages called attractor points. The ratio masks are calculated based on the distances between the embeddings and the attractor points. Minimizing the estimation error of the separated sources makes the embeddings close to the corresponding attractor points. Unlike PIT and typical mask-based methods, an arbitrary number of sources can be dealt with in the inference phase regardless of the number of sources configured in the training phase. Deep clustering (DC) [54, 55] is similar to the DAN in a way that the embedding is calculated for each TF bin. Assuming that each TF bin corresponds to one sound source, a DNN is trained such that the embeddings of the same source become close to each other and the embeddings of different sources become orthogonal. In the inference phase, for calculating the binary masks of each source, the embeddings are clustered into an arbitrary number of sources with the K-means algorithm.

To avoid using the phase information, several studies focus on time-domain separation in exchange for the degraded quality of separated speech under the low SNR condition. The speech enhancement generative adversarial network (SEGAN) [61] consists of a generator network that estimates clean speech from noisy speech and a discriminator network that tries to detect the generated speech

as a fake. Unlike the standard GAN, paired data should be used for stabilizing the training. The fully-convolutional time-domain audio separation network (Conv-TasNet) [62, 63] based on an encoder-decoder model achieved the state-of-the-art performance of single-channel source separation. First, the encoder transforms a time-domain mixture signal into a spectrogram-like representation. The masks for each source are then estimated and the spectrogram-like representation of each source is obtained by using the estimated masks. Finally, the decoder recovers the time-domain source signal. This model is trained such that the scale-invariant signal-to-noise ratios (SI-SNRs) [65] of the estimated source signals are maximized. Although these supervised methods work well in known environments, they often fail to generalize to unseen environments [33, 34, 66].

Recently, deep generative models of speech spectra based on variational autoencoders (VAEs) have been used for semi-blind single-channel speech enhancement. Bando *et al.* [40] first proposed a unified model that consists of an NMF-based source model for noise (noise model) and a DNN-based one for speech (speech model) with latent variables. The speech model is given as the decoder of a VAE trained beforehand from clean speech data in an unsupervised manner. On the other hand, the noise model is optimized on-the-fly for observed noisy speech data. This approach mitigates the sensitivity to the acoustic characteristics of noisy environments. Leglaive *et al.* [41] proposed a similar model for maximum likelihood estimation, while [40] is based on a Bayesian inference.

2.1.3 Unsupervised Multichannel Blind Methods

We here mainly focus on multichannel blind source separation (BSS) based on unsupervised learning of a probabilistic model consisting of a spatial model and a source model. We first introduce BSS methods based on the rank-1 spatial model, and then introduce BSS methods based on the full-rank spatial model.

Rank-1 Spatial Model

The methods below assume the time-invariant linear mixing system given by Eq. (1.3). Frequency-domain independent component analysis (ICA) [22,67,68] is the most basic unsupervised BSS method based on the independence of sources. This method can be used under a determined condition, where the number of microphones M is equal to the number of sound sources N. A determined mixing process given by Eq. (1.4) enables us to consider its inverse process called a demixing process given as

$$\mathbf{y}_{ft} \triangleq \mathbf{D}_f \mathbf{x}_{ft}, \tag{2.10}$$

where $\mathbf{y}_{ft} \triangleq [y_{1ft}, \ldots, y_{Mft}] \in \mathbb{C}^M$ and $\mathbf{D}_f \triangleq \mathbf{A}_f^{-1}$ is called a demixing matrix. The goal of ICA is thus to estimate frequency-wise demixing matrices $\mathbf{D}_f$ such that $\{y_{mft}\}_{m=1}^M$ become independent. Note that ICA is considered to have a simple source model given by Eq. (1.2) that assumes that the TF bins of each source independently follow *univariate* complex non-Gaussian distributions. This causes the permutation problem because all frequency bins are processed independently.

To solve this problem, independent vector analysis (IVA) [23, 69] based on a modified source model assumes that the time frames of each source, $\mathbf{s}_{n:t} \triangleq [s_{n1t}, \ldots, s_{nFt}] \in \mathbb{C}^F$, follow complex *multivariate* generalized Gaussian distributions such as Laplace and Gaussian distributions. To accelerate and stabilize IVA, Ono [70] proposed a convergence-guaranteed parameter estimation method called iterative projection (IP). Recently, IP method has been further extended to improve the convergence speed and separation performance [71–73]. IVA has been extended for dealing with an overdetermined condition ($N < M$) [74, 75], where $M - N$ sources of no interest in addition to N sources are internally considered to recover a determined condition. Nugraha *et al.* [76] proposed normalizing flow (NF)-IVA that uses a time-varying linear transformation based on an NF instead of using a time-invariant linear transformation.

To further mitigate the permutation problem left in IVA, Kitamura *et al.* [28] proposed independent low-rank matrix analysis (ILRMA) based on a low-rank source model that assumes the TF bins of each source to follow *univariate* complex

Gaussian distributions with the PSDs factorized by NMF as

$$s_{nft} \sim \mathcal{N}_{\mathbb{C}} \left(0, \sum_{k=1}^{K} w_{nkf} h_{nkt} \right). \tag{2.11}$$

Using Eq. (1.3), a TF bin of a source image, $\mathbf{x}_{nft}$, can be said to follow a *degenerate multivariate* complex Gaussian distribution with a rank-1 SCM as follows:

$$\mathbf{x}_{nft} \sim \mathcal{N}_{\mathbb{C}} \left(\mathbf{0}_M, \left(\sum_{k=1}^{K} w_{nkf} h_{nkt} \right) \mathbf{G}_{nf} \right), \tag{2.12}$$

where $\mathbf{G}_{nf} \triangleq \mathbf{a}_{nf}\mathbf{a}_{nf}^{\mathsf{H}}$ is the rank-1 SCM. Using Eq. (1.1) and the additive property of the Gaussian distribution, the observed spectrum $\mathbf{x}_{ft}$ is given by

$$\mathbf{x}_{ft} \sim \mathcal{N}_{\mathbb{C}} \left(\mathbf{0}_M, \sum_{n=1}^{N} \left(\sum_{k=1}^{K} w_{nkf} h_{nkt} \right) \mathbf{G}_{nf} \right). \tag{2.13}$$

For maximum-likelihood estimation of the parameters, an efficient convergence-guaranteed optimization algorithm iterating IS-NMF and IP was derived. In spite of the severely restricted ability of the rank-1 spatial model, ILRMA is empirically known to work stably in real environments. Independent positive semidefinite tensor analysis (IPSDTA) [77] is an extension of ILRMA and uses a PSDTF-based source model instead of the NMF-based source model.

Several attempts have been made to enable ILRMA to deal with an overdetermined condition ($N < M$). Kitamura *et al.* [78] used ILRMA with M microphones for estimating M *components* clustered into N sources. Note that the component-source association should be specified in advance. In practice, $M = NP$ should be required for stable estimation, where P is the number of components associated to each source and represents the rank of the SCMs of the source. If $P = 2$, for example, each source would be represented by two components corresponding to direct and reflective propagation paths (multi-modal directivity). Kubo *et al.* [79] used ILRMA for speech enhancement. Specifically, the rank-1 SCMs of directional speech and the rank-$(M-1)$ SCMs of diffuse noise are estimated with ILRMA and the missing rank-1 SCMs and PSDs of speech and noise are then estimated in an independent step.

In the above methods, the SCMs of all sources are assumed to be rank-1 matrices. This assumption indicates that the sound propagation process is time-invariant, and the reverberation is so short that the reverberation of a certain time frame does not affect the observation of other time frames. However, the rank-1 assumption often does not hold in a real environment.

Full-Rank Spatial Model

Duong *et al.* [24] pioneered a BSS method based on the full-rank spatial model, which was called full-rank spatial covariance analysis (FCA) in [35, 80]. In theory, BSS methods based on full-rank spatial models can be used under either of determined ($N = M$), overdetermined ($M > N$), and underdetermined ($M < N$) conditions. In particular, the overdetermined condition is considered as the most important because it is often the case that at most two or three sources of interest are overlapped. From a practical point of view, more sources are considered to be hard to separate reasonably. The full-rank spatial model can represent an *echoic* sound propagation process, and each bin of each source image is assumed to follow a *multivariate* complex Gaussian distribution with a full-rank SCM, that is, $\mathbf{G}_{nf}$ in Eq. (2.13) is assumed to be a full-rank matrix.

Because FCA has no specific source model, it suffers from the permutation problem like ICA. To alleviate this problem, multichannel NMF (MNMF) based on the low-rank source model given by Eq. (2.11) has been developed [25–27]. The first formulation of MNMF was proposed by Ozerov *et al.* [25], where full-rank noise SCMs and rank-1 source SCMs are used and the cost function based on the Itakura-Saito (IS) divergence is minimized by using a multiplicative update or expectation-maximization (EM) algorithm. This method was extended such that all sources have full-rank SCMs [26]. Sawada *et al.* [27] introduced a partitioning function to share a set of basis spectra by all sources and derived a majorization-minimization (MM) algorithm. Nikunen and Virtanen [81, 82] proposed a model similar to [27] which represents the SCM of each source as the weighted sum of all possible direction-dependent SCMs. ILRMA was originally derived by integrating the low-rank source model into IVA and was shown to be

a special case of MNMF obtained by restricting the SCMs of sources to rank-1 matrices. Although a convergence-guaranteed closed-form iterative optimization algorithm has been developed for MNMF [27], it tends to easily get stuck at bad local optima because of the strong initialization sensitivity and suffers from the high computational cost because of the repeated heavy matrix operations.

The joint diagonalization of covariance matrices for accelerated computation has gained much attention in recent years. For multichannel BSS, Ito and Nakatani [35, 80] proposed a fast version of FCA called FastFCA that restricts the SCMs of sources to jointly-diagonalizable (JD) yet full-rank matrices as follows:

$$\forall n, \quad \mathbf{G}_{nf} = \mathbf{Q}_f^{-1}\mathrm{Diag}(\tilde{\mathbf{g}}_{nf})\,\mathbf{Q}_f^{-\mathsf{H}}, \tag{2.14}$$

where $\tilde{\mathbf{g}}_{nf} = [\tilde{g}_{nf1}, \ldots, \tilde{g}_{nfM}] \in \mathbb{R}_+^M$ is a nonnegative vector and $\mathbf{Q}_f \in \mathbb{C}^{F\times F}$ is a non-singular matrix called *diagonalizer*. In [35], the number of sound sources was limited to two because two positive definite matrices are mathematically JD; the joint-diagonalization constraint on two SCMs does not change the degree of freedom. In [80], the joint-diagonalization constraint was applied to the SCMs of an arbitrary number of sound sources at the expense of the degree of freedom. Then, we [83] and Ito and Nakatani [44] proposed a fast version of MNMF called FastMNMF1 independently and concurrently. As discussed in Section 2.1.1, for single-channel BSS, Yoshii *et al.* [51] proposed a fast version of CTF [49] called ILRTA (a.k.a. FastCTF) that restricts frequency and time covariance matrices to JD full-rank matrices, independently and concurrently with FastFCA [35]. To estimate the diagonalizer $\mathbf{Q}_f$, a convergence-guaranteed IP method was used in [83] as in ILRTA [51], while a fixed point iteration (FPI) method without convergence guarantee was used in [35, 44, 80]. In Chapter 4, we further extend FastMNMF1 to reduce the initialization sensitivity of FastMNMF1.

Several studies use fixed diagonalizers for efficient source separation in a transformed space [84–87]. Lee *et al.* [84], for example, used as the diagonalizer a beamspace transform matrix calculated from premeasured steering vectors. Mitsufuji *et al.* [85] proposed a variant of FastMNMF fixing the diagonalizer to the discrete Fourier transform (DFT) matrix for a straight-shape array of a large

number (e.g., 32) of equally-spanned microphones. To relax this condition, the steering vectors of all possible directions were used in [87]. Taniguchi *et al.* [86] proposed a prototype of FastMNMF and found that a demixing matrix estimated by IVA works better as the diagonalizer than the beamspace transform matrix.

While the methods discussed above assume that the mixture spectrograms are sum of the source spectrograms, complex Gaussian mixture model (cGMM)-based source separation methods [30, 36, 88–90] assume that the source spectrograms are sparse and only one of the sources is dominant in each TF bin. Thus, the observed spectrum $\mathbf{x}_{ft}$ is given by

$$\mathbf{x}_{ft} \sim \prod_{n=1}^{N} \mathcal{N}_{\mathbb{C}} \left(\mathbf{0}_M, \lambda_{nft}\mathbf{G}_{nf}\right)^{z_{ftn}}, \tag{2.15}$$

where z_{ftn} indicates whether source n is dominant or not at frequency f and time frame t. Since all frequency bins are processed independently in cGMM, it suffers from the permutation problem as ICA and FCA. In [30, 88], assuming that the steering vectors of all possible directions are known, the directions of all sources are estimated and used for solving the permutation problem. Alternatively, as MNMF, Itakura *et al.* [90] introduced the NMF-based source model into cGMM. When background noise exists, the assumption that only one source is dominant in each TF bin does not hold, and the performance drastically degrades. To solve this problem, Ito *et al.* [36] proposed noisy-cGMM that assumes that each TF bin consists of one source and noise signals as follows:

$$\mathbf{x}_{ft} \sim \prod_{n=1}^{N} \mathcal{N}_{\mathbb{C}} \left(\mathbf{0}_M, \lambda_{nft}\mathbf{G}_{nf} + \lambda_{0ft}\mathbf{G}_{0f}\right)^{z_{ftn}}, \tag{2.16}$$

where λ_{0ft} and $\mathbf{G}_{0f}$ are the PSD and SCM of the noise source. Moreover, the joint diagonalization constraint on the SCMs is used to reduce the computational cost. The relationship of the multichannel source separation methods discussed in this section is summarized in Table 2.1.

2.1.4 Multichannel Methods Based on Deep Learning

A typical approach to integrating a DNN with multichannel source separation or speech enhancement is to use beamforming such as minimum variance

Table 2.1: The multichannel source separation methods based on a probabilistic generative model. Note that non-blind methods are not exactly based on the generative model.

	Rank-1	JD full-rank	Full-rank
Blind	ICA [22, 67, 68], IVA [23, 69–73], ILRMA [28], IPSDTA [77], OverIVA [74,75], NF-IVA [76]	Fixed diagonalizers + MNMF [84–87], FastFCA [35, 80], FastMNMF1 [44,83], FastMNMF2 [43]	FCA [24], MNMF [25–27, 81, 82], cGMM [30, 88, 89], cGMM+NMF [90], Noisy-cGMM [36]
Semi-blind	ILRMA-DSP [42], MVAE [91]	FastMNMF-DSP [83]	MNMF-DSP [37,42,92, 93], GMVAE [45, 94]
Non-blind	(IDLMA [95])		(Full-rank+DAE [96])

distortionless response (MVDR) [97] and generalized eigenvalue (GEV) [98] beamforming. Before the emergence of deep learning, assuming that the steering vectors of all possible directions are known, beamforming had widely been used along with source localization methods such as multiple signal classification (MUSIC) [99]. Instead of using the steering vectors prepared in advance, time-frequency masks are estimated using a DNN trained with paired data and then used for calculating the SCMs of speech and noise [11, 12] as follows:

$$\mathbf{G}_{nf} = \sum_{t=1}^{T} m_{nft} \mathbf{x}_{ft} \mathbf{x}_{ft}^{\mathsf{H}}, \tag{2.17}$$

where m_{nft} is the TF mask for source n (speech or noise) at frequency f estimated by the DNN, and the steering vector of source n is given as an eigenvector corresponding to the first principal component of $\mathbf{G}_{nf}$. The DNN can be trained without using paired data [100–104]. It can also be trained jointly with an ASR loss [100–102]. Specifically, beamforming filters are calculated using a DNN, and the beamformed signals are fed into an ASR network. The ASR loss is back-propagated to train the DNN for estimating beamforming filters. In [103, 104], the cost function derived from multichannel source separation methods are used for training the DNN. In [95, 96], a DAE is integrated into a process of SCM-based multichannel source separation; (1) the observed mixture spectra are separated

into speech and noise by using the current estimate of the speech and noise SCMs, (2) the PSDs of the enhanced speech are further refined by using the DAE, and (3) the speech and noise SCMs are updated by using the current estimate of the PSDs of the speech and noise. [96] is based on a full-rank SCM, and [95] is based on a rank-1 SCM and is called independent deeply learned matrix analysis (IDLMA). Although this method is similar to the methods described in Section 3 in that both methods iteratively optimize the SCMs and PSDs using DNNs, this approach does not guarantee the monotonic increase of the likelihood. Such supervised methods are known to work well in a known environment, but adaptation to unseen noisy environments is still an open problem.

The semi-blind single-channel speech enhancement method [40] that uses a DNN-based speech model has been extended for semi-blind multichannel speech enhancement [37, 42, 83, 92] and speech separation [45, 91, 94]. As in [40], the speech enhancement methods use a DNN-based speech model and an NMF-based noise model with the full-rank [37, 92], rank-1 [42], and JD full-rank spatial model [83]. In [93], a variant of NF called the generative flow (GF) is used for DNN-based speech model instead of VAE in [37, 40, 42, 83, 92]. For speech separation, DNN-based speech models based on a conditional VAE (CVAE) [105] that uses utterance-wise speaker labels for training are used for all sources with the rank-1 [91] and full-rank spatial model [45, 94]. In [94], frame-wise phonetic labels are used in addition to the utterance-wise speaker label for the CVAE.

2.2 Dereverberation

We categorize dereverberation methods and joint source separation and dereverberation methods into unsupervised blind and supervised non-blind methods.

2.2.1 Unsupervised Blind Methods

Reverberation has typically been represented with a moving average (MA) model and/or an autoregressive (AR) model. In [15–17], using a time-invariant MA model, the direct signal is obtained with spectral subtraction (SS), where only the

power spectra are considered and the phase information is discarded. In [106], using a time-varying MA model, the direct signal is estimated with the EM algorithm and the Kalman filter.

Linear prediction (LP) and its multichannel extension (MCLP) have been used in the time or frequency domain [19, 107–113]. In frequency-domain MCLP, for example, reverberations are represented with an AR model as follows:

$$x_{ftm} = d_{ftm} + \sum_{l=1}^{L} \mathbf{b}_{flm} \mathbf{x}_{f,t-l}, \tag{2.18}$$

where d_{ftm} is the direct signal recorded by m-th microphone and $\mathbf{b}_{flm}$ is the AR coefficients. In terms of linear prediction, d_{ftm} is regarded as the prediction error. The AR coefficients are estimated such that the mean squared error is minimized. This is equivalent to the maximum likelihood estimation based on the assumption that d_{ftm} is Gaussian white noise. Because of this assumption, when this method is used for speech dereverberation, the estimated speech spectra tend to be white. To alleviate the problem, various approaches have been proposed [19, 108, 111, 112, 114].

The weighted prediction error (WPE) [19, 113] is one of the most successful dereverberation methods and has been used in commercial devices such as smart speakers [3, 4]. It introduces the Gaussian source model given by Eq. (1.2) and the delay parameter Δ as follows:

$$x_{ftm} = d_{ftm} + \sum_{l=\Delta}^{L} \mathbf{b}_{flm} \mathbf{x}_{f,t-l}. \tag{2.19}$$

The reverberation is divided into two parts: early reflection and late reverberation, and the latter is known to be more harmful to the speech intelligibility and ASR performance [115,116]. The delay parameter is effective for removing only the late reverberation without whitening the target signal. The PSDs of the target signal and the AR coefficients are iteratively and alternately updated until convergenece. In [117], the time-varying AR coefficients are used to deal with moving sources and the parameters are estimated efficiently using the Kalman filter.

For joint *blind* source separation and dereverberation, autoregressive ILRMA (AR-ILRMA) [118] that combines ILRMA [28] based on the rank-1 spatial model

with the AR-based reverberation model given by Eq. (2.19) [19,113] was proposed. In [119], an SCM-based BSS method called full-rank covariance analysis (FCA) [24] based on the full-rank spatial model was integrated with an autoregressive moving average (ARMA)-based reverberation model (called ARMA-FCA in this paper). Although ARMA-FCA can deal with diffuse noise thanks to the full-rank spatial model, it needs to solve the permutation problem in a post-processing step because of the frequency-wise source separation. To avoid the permutation problem under a determined condition, in [120], the permutation problem of ARMA-FCA was alleviated by utilizing the parameters estimated by AR-ILRMA. The computational cost of ARMA-FCA, however, is larger than those of AR-ILRMA because of the unconstrained full-rank SCMs.

2.2.2 Supervised Non-blind Methods

A typical approach for supervised dereverberation is to estimate the magnitude spectrogram directly [121–123] or TF-masks [124] of the direct signal. In [122], a DNN is trained such that the noisy reverberant spectrograms are mapped to the corresponding clean spectrograms. In [123], a fully-convolutional network (FCN) is used to capture time-frequency structures of speech and is trained using a GAN-based approach.

WPE is also used in supervised approaches. In [32,125], to avoid the iterative update of WPE, a DNN is used for estimating the PSDs of the direct signal given the observed reverberant signals. For joint speech dereverberation and enhancement, one can sequentially use WPE and beamforming based on deep neural networks (DNNs) [126], where the time-frequency (TF) masks of direct speech are estimated with a DNN for calculating the dereverberation filters [32, 125] and those of speech and noise are then estimated from the dereverberated signals with another DNN for calculating demixing filters [11,12]. While these DNNs are concatenated and jointly optimized in the training phase such that the ASR performance for the dereverberated enhanced speech is maximized, such a supervised approach increases the sensitivity to the environment. In the test phase, WPE and DNN-based beamforming can be used alternately

and iteratively [127]. Extending this approach to multiple speech separation under a condition that the TF-masks of each source are given, a joint separation, dereverberation, and denoising method was proposed [128]. Although DNN-based mask estimation is computationally efficient, robust mask estimation from noisy reverberant mixture signals is still an open problem because the acoustic characteristics of a real environment may significantly differ from those covered by the training data.

Chapter 3

Semi-blind Multichannel Speech Enhancement Based on a Deep Generative Source Model

3.1 Introduction

Speech enhancement plays a vital role for automatic speech recognition (ASR) in noisy environments. Although the performance and robustness of ASR have been drastically improved thanks to the development of deep learning techniques, ASR in unseen noisy environments that are not covered by training data is still an open problem. Many methods have thus been proposed for single-channel or multichannel speech enhancement. These methods can be categorized into supervised, semi-blind, and blind methods.

A popular approach to supervised speech enhancement is to train deep neural networks (DNNs) by using pairs of noisy and clean speech signals. In single-channel speech enhancement, one can use denoising autoencoders (DAEs) that take noisy speech spectra as input, and output clean speech spectra [9]. Alternatively, DNNs can be trained to estimate time-frequency masks, i.e., classify each time-frequency bin into speech or noise [10, 13]. In multichannel speech enhancement using phase information, the estimated masks are used for calculating the spatial covariance matrices (SCMs) of speech and noise. This allows one to use beamforming methods [11, 12]. Although this approach has successfully been used as a front end of ASR, the performance of speech

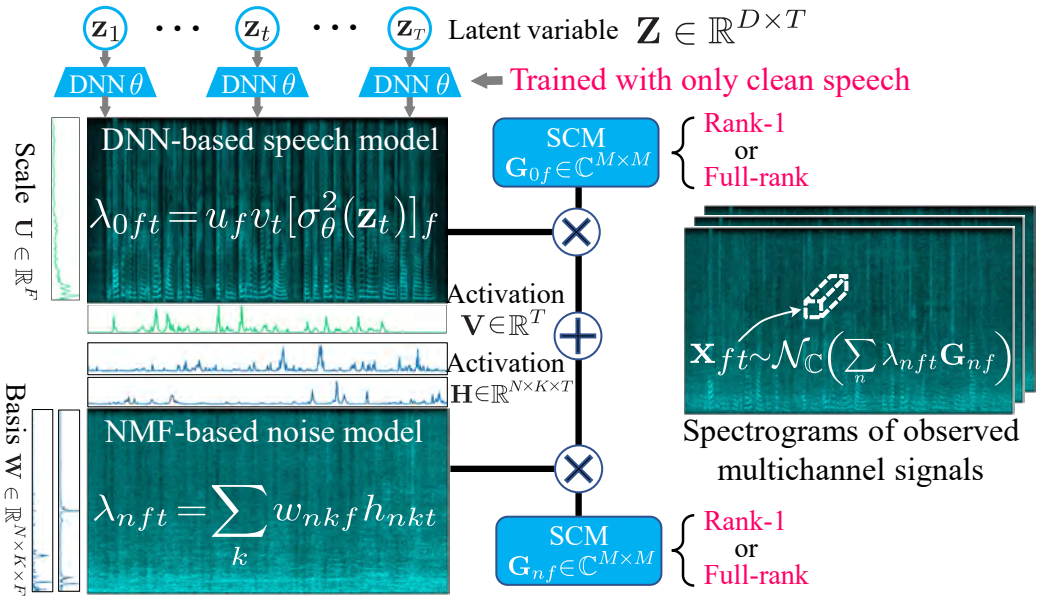


Figure 3.1: A probabilistic generative model of multichannel noisy speech spectra based on a deep generative model of speech, a low-rank model of noise, and a full-rank spatial model.

enhancement is often considerably degraded in unseen noisy environments due to the nature of supervised mask estimation [33].

To mitigate the sensitivity to acoustic characteristics of noisy environments, one may use blind methods such as multichannel extensions of nonnegative matrix factorization (NMF) [25–28, 81, 90]. Each variant of multichannel NMF (MNMF) is based on a probabilistic generative model of the complex spectrograms of mixture signals consisting of a source and spatial model and is used for general blind sound separation (BSS). The key assumption underlying the family of MNMF is that the power spectral densities (PSDs) of all sound sources have low-rank structure. In speech enhancement, however, the performance of MNMF is limited because the low-rank assumption does not hold for the PSDs of speech. Several studies thus integrated a DAE into an optimization step of MNMF which estimates the PSDs of speech [95, 96]. Although such integration of a powerful DNN and a physically founded statistical model is promising, supervised learning of DAEs causes sensitivity to noisy environments again.

To solve the problems of the conventional DNN- and MNMF-based methods,

we propose a semi-blind method that uses a pretrained generative model of natural speech (speech model). More specifically, we formulate a DNN-based speech model called a deep speech prior that represents the generative process of the complicated PSDs of clean speech [40] and an NMF-based noise model that represents the generative process of the low-rank PSDs of noise. A unified generative model of observed noisy speech is then obtained by integrating those source models with a full-rank or rank-1 spatial model as in MNMF [27] or its constrained version called independent low-rank matrix analysis (ILRMA) [28], respectively.

As the deep speech *prior*, we formulate a latent variable model that implicitly represents the time-frequency features of speech spectra including but not limited to fundamental frequencies (F0s), harmonic structures, and spectral envelopes. To achieve this, the parameters of this model are learned from clean speech data in an unsupervised variational auto-encoding manner. The noise model, in contrast, is learned on-the-fly without pre-training. Given noisy speech as observed data, the latent variables of the speech model, the full-rank or rank-1 SCMs and PSDs of speech and noise can be estimated in an unsupervised maximum-likelihood manner by combining a majorization-minimization algorithm with Metropolis sampling or backpropagation. Finally, the *posterior* of clean speech spectra can be computed via multichannel Wiener filtering.

In this paper, the *deep speech prior* refers to a DNN-based generative model of natural speech spectra, which can be used as a prior for speech enhancement. It sounds similar to the so-called *deep image prior* [129], which refers to a DNN-based generative model of natural images. The DNN of the speech prior can take any architecture and needs to be trained from speech data such that it properly represents a probability distribution of natural speech spectra. In contrast, the deep image prior is based on a deep convolutional architecture, and the network architecture itself is found to work as an inductive bias for generating natural images without any training. Investigation of such an inductive bias that encourages the DNN to generate natural speech spectra is an interesting future direction.

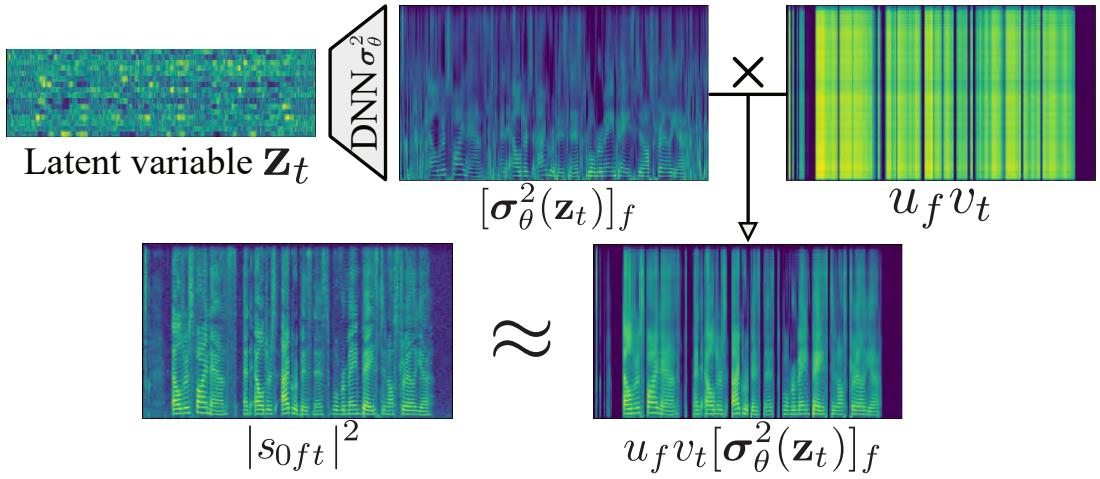


Figure 3.2: The proposed DNN-based speech model. The PSDs $\{\lambda_{0ft}\}_{f=1}^F$ of a speech spectrum $\{s_{0ft}\}_{f=1}^F$ at time t are obtained by feeding the latent variable $\mathbf{z}_t$ following the standard Gaussian distribution into a DNN $\boldsymbol{\sigma}_\theta^2$ with parameters θ and then scaling the output $\boldsymbol{\sigma}_\theta^2(\mathbf{z}_t)$ according to u_f and v_t.

3.1.1 DNN-Based Speech Model

As in Fig. 3.2, the PSD of the source n at frequency f and time t is determined by a DNN as follows:

$$\lambda_{nft} = u_f v_t [\boldsymbol{\sigma}_{\boldsymbol{\theta}}^2(\mathbf{z}_t)]_f, \tag{3.1}$$

where $\boldsymbol{\sigma}_{\boldsymbol{\theta}}^2(\cdot)$ is a nonlinear function (DNN) with parameters $\boldsymbol{\theta}$ that maps a D-dimensional real vector $\mathbf{z}_t \in \mathbb{R}^D$ to an F-dimensional nonnegative vector $\boldsymbol{\sigma}_{\boldsymbol{\theta}}^2(\mathbf{z}_t) \in \mathbb{R}_+^F$, $[\cdot]_f$ indicates the f-th element of a vector, $u_f \geq 0$ is a scaling factor at frequency f, and $v_t \geq 0$ is an activation at time t. We assume that $\mathbf{z}_t$ follows a standard Gaussian distribution as follows:

$$\mathbf{z}_t \sim \mathcal{N}(\mathbf{0}_D, \mathbf{I}_D), \tag{3.2}$$

where $\mathbf{0}_D$ and $\mathbf{I}_D$ are the all-zero vector and the identity matrix of size D, respectively. $\mathbf{z}_t$ implicitly represents the characteristics (e.g., fundamental frequencies (F0s), harmonic structures, and formants) of the PSDs $\{\lambda_{nft}\}_{f=1}^F$ of the speech at time t. While the DNN specified by $\boldsymbol{\theta}$ is trained from clean speech data (Section 3.4.3), the latent variables $\mathbf{Z} \triangleq \{\mathbf{z}_t\}_{t=1}^T$ are estimated on-the-fly. The scaling

factors $\mathbf{U} \triangleq \{u_f\}_{f=1}^{F}$ and the activations $\mathbf{V} \triangleq \{v_t\}_{t=1}^{T}$ are introduced for resolving the scale ambiguity of model parameters.

3.1.2 NMF-Based Noise Model

The PSD of source n at frequency f and time t is represented in the framework of NMF as follows:

$$\lambda_{nft} = \sum_{k=1}^{K} w_{nkf} h_{nkt}, \tag{3.3}$$

where K denotes the number of bases, $w_{nkf} \geq 0$ indicates the magnitude of basis k of source n at frequency f, and $h_{nkt} \geq 0$ indicates the activation of basis k of source n at time t. $\mathbf{W} \triangleq \{w_{nkf}\}_{n=1,k=1,f=1}^{N,K,F}$ and $\mathbf{H} \triangleq \{h_{nkt}\}_{n=1,k=1,t=1}^{N,K,T}$ are estimated on-the-fly.

3.2 MNMF with a Deep Speech Prior (MNMF-DSP)

We formulate a unified probabilistic generative model by integrating the DNN-based speech model, NMF-based noise model, and the full-rank spatial model described in Section 1.3.2 and then derive an update rule based on a minorization-maximization (MM) algorithm.

3.2.1 Formulation

In this section, we assume source 0 corresponds to a speech source and source n (≥ 1) corresponds to a noise source ($N+1$ sources in total). Substituting Eq. (3.1) and Eq. (3.3) into Eq. (1.6), we obtain the likelihood function of unknown variables $\mathbf{Z}$, $\mathbf{U}$, $\mathbf{V}$, $\mathbf{W}$, $\mathbf{H}$, and $\mathbf{G}$ as follows:

$$\begin{aligned}
&\log p(\mathbf{X}|\mathbf{Z}, \mathbf{U}, \mathbf{V}, \mathbf{W}, \mathbf{H}, \mathbf{G}) \\
&= \sum_{f=1}^{F} \sum_{t=1}^{T} \log \mathcal{N}_{\mathbb{C}} \left(\mathbf{x}_{ft} | \mathbf{0}_M, \mathbf{Y}_{ft}\right) &(3.4)\\
&= \sum_{f=1}^{F} \sum_{t=1}^{T} \left(-\mathrm{tr}\left(\mathbf{Y}_{ft}^{-1} \mathbf{X}_{ft}\right) - \log |\mathbf{Y}_{ft}|\right) + \mathrm{const}, &(3.5)
\end{aligned}$$

where $\mathbf{X}_{ft} \in \mathbb{S}_+^M$ and $\mathbf{Y}_{ft} \in \mathbb{S}_+^M$ are given by

$$\mathbf{X}_{ft} \triangleq \mathbf{x}_{ft}\mathbf{x}_{ft}^{\mathsf{H}}, \tag{3.6}$$

$$\mathbf{Y}_{ft} \triangleq \sum_{n=0}^{N} \lambda_{nft}\mathbf{G}_{nf}. \tag{3.7}$$

λ_{0ft} and $\lambda_{nft}(n \geq 1)$ are the PSD of the speech and that of the noise, respectively, which are given by

$$\lambda_{nft} = \begin{cases} u_f v_t[\boldsymbol{\sigma}_{\boldsymbol{\theta}}^2(\mathbf{z}_t)]_f & (n = 0), \\ \sum_{k=1}^{K} w_{nkf}h_{nkt} & (n \geq 1). \end{cases} \tag{3.8}$$

We define $\mathbf{Y}_{nft} \triangleq \lambda_{nft}\mathbf{G}_{nf}$ and $\mathbf{Y}_{n(\geq 1)ftk} \triangleq w_{nkf}h_{nkt}\mathbf{G}_{nf}$.

Our goal is to estimate $\mathbf{Z}$, $\mathbf{U}$, $\mathbf{V}$, $\mathbf{W}$, $\mathbf{H}$, and $\mathbf{G}$ such that the log-likelihood $\log p(\mathbf{X}|\mathbf{Z}, \mathbf{U}, \mathbf{V}, \mathbf{W}, \mathbf{H}, \mathbf{G})$ given by Eq. (3.5) is maximized. To avoid the scale ambiguity of the parameters, we put normalization constraints on $\mathbf{U}$, $\mathbf{W}$, and $\mathbf{G}$ as follows:

$$\sum_{f=1}^{F} u_f = 1, \tag{3.9}$$

$$\sum_{f=1}^{F} w_{nkf} = 1, \tag{3.10}$$

$$\mathrm{tr}(\mathbf{G}_{nf}) = 1. \tag{3.11}$$

3.2.2 Optimization

We aim to estimate the parameters $\mathbf{Z}$, $\mathbf{U}$, $\mathbf{V}$, $\mathbf{W}$, $\mathbf{H}$, and $\mathbf{G}$ that maximize $\log p(\mathbf{X}|\mathbf{Z}, \mathbf{U}, \mathbf{V}, \mathbf{W}, \mathbf{H}, \mathbf{G})$ given by Eq. (3.5) and obtain the enhanced speech image $\mathbf{x}_{0ft}^{\mathrm{FR}} \in \mathbb{C}^M$ using a multichannel Wiener filter (MWF) given by Eq.(1.7). Since it is hard to directly maximize the log-likelihood with respect to each of these parameters, we use an MM algorithm that iteratively maximizes lower bounds of the log-likelihood as in MNMF [27].

Matrix Inequalities

To derive the lower bounds, we use two matrix inequalities on positive definite matrices [90]. For a convex function $f_1(\mathbf{S}) = -\log|\mathbf{S}|$ with respect to $\mathbf{S} \in \mathbb{S}_+^M$,

we calculate a tangent plane at an arbitrary point $\mathbf{\Omega} \in \mathbb{S}_+^M$ by using a first-order Taylor expansion as follows:

$$-\log|\mathbf{S}| \geq -\log|\mathbf{\Omega}| - \mathrm{tr}\big(\mathbf{\Omega}^{-1}\mathbf{S}\big) + M, \tag{3.12}$$

where the equality holds if and only if $\mathbf{\Omega} = \mathbf{S}$. For a concave function $f_2(\mathbf{S}) = -\mathrm{tr}(\mathbf{S}^{-1}\mathbf{R})$ with any matrix $\mathbf{R} \in \mathbb{S}_+^M$ with respect to $\mathbf{S} \in \mathbb{S}_+^M$, we have

$$-\mathrm{tr}\left(\left(\sum_{k=1}^{K}\mathbf{S}_k\right)^{-1}\mathbf{R}\right) \geq -\sum_{k=1}^{K}\mathrm{tr}\big(\mathbf{S}_k^{-1}\mathbf{\Phi}_k\mathbf{R}\mathbf{\Phi}_k^{\mathsf{H}}\big), \tag{3.13}$$

where $\{\mathbf{S}_k\}_{k=1}^K$ ($\mathbf{S}_k \in \mathbb{S}_+^M$) is a set of positive definite matrices, $\{\mathbf{\Phi}_k\}_{k=1}^K$ is a set of auxiliary matrices that sum to the identity matrix, i.e., $\sum_{k=1}^K \mathbf{\Phi}_k = \mathbf{I}_M$, and the equality holds if and only if $\mathbf{\Phi}_k = \mathbf{S}_k(\sum_{k'=1}^K \mathbf{S}_{k'})^{-1}$.

Deriving Lower Bounds

Using Eqs. (3.12) and (3.13) and introducing auxiliary matrices $\mathbf{\Omega} \triangleq \{\mathbf{\Omega}_{ft}\}_{f,t=1}^{F,T}$ and $\mathbf{\Phi} \triangleq \{\mathbf{\Phi}_{0ft}\}_{f,t=1}^{F,T} \cup \{\mathbf{\Phi}_{nft}\}_{n,f,t=1}^{N,F,T}$, we can derive a lower bound of Eq. (3.5), $\mathcal{L}_{\mathrm{FR}}^1(\mathbf{Z},\mathbf{U},\mathbf{V},\mathbf{W},\mathbf{H},\mathbf{G},\mathbf{\Omega},\mathbf{\Phi})$, as follows:

$$\begin{aligned}
&\log p(\mathbf{X}|\mathbf{Z},\mathbf{U},\mathbf{V},\mathbf{W},\mathbf{H},\mathbf{G}) \\
&\geq -\sum_{f=1}^{F}\sum_{t=1}^{T}\sum_{n=0}^{N}\lambda_{nft}^{-1}\mathrm{tr}\big(\mathbf{G}_{nf}^{-1}\mathbf{\Phi}_{nft}\mathbf{X}_{ft}\mathbf{\Phi}_{nft}^{\mathsf{H}}\big) \\
&\quad -\sum_{f=1}^{F}\sum_{t=1}^{T}\sum_{n=0}^{N}\lambda_{nft}\mathrm{tr}\big(\mathbf{G}_{nf}\mathbf{\Omega}_{ft}^{-1}\big) - \sum_{f=1}^{F}\sum_{t=1}^{T}\log|\mathbf{\Omega}_{ft}| + \mathrm{const} &(3.14)\\
&\triangleq \mathcal{L}_{\mathrm{FR}}^1(\mathbf{Z},\mathbf{U},\mathbf{V},\mathbf{W},\mathbf{H},\mathbf{G},\mathbf{\Omega},\mathbf{\Phi}), &(3.15)
\end{aligned}$$

where the equality holds if and only if

$$\mathbf{\Omega}_{ft} = \mathbf{Y}_{ft}, \tag{3.16}$$

$$\mathbf{\Phi}_{nft} = \mathbf{Y}_{nft}\mathbf{Y}_{ft}^{-1}. \tag{3.17}$$

Note that $\mathbf{Y}_{ft} \triangleq \sum_{n=0}^N \mathbf{Y}_{nft}$ and $\mathbf{Y}_{nft} \triangleq \lambda_{nft}\mathbf{G}_{nf}$.

Using $\mathbf{\Omega}$ and $\mathbf{\Phi}$ and introducing additional auxiliary matrices $\mathbf{\Psi} = \{\mathbf{\Psi}_{nftk}\}_{f,t,n,k=1}^{F,T,N,K}$, we can derive another lower bound , $\mathcal{L}_{\mathrm{FR}}^2(\mathbf{Z},\mathbf{U},\mathbf{V},\mathbf{W},\mathbf{H},\mathbf{G},\mathbf{\Omega},\mathbf{\Phi},\mathbf{\Psi})$ of Eq. (3.5)

as follows:

$$\begin{aligned}
&\log p(\mathbf{X}|\mathbf{Z},\mathbf{U},\mathbf{V},\mathbf{W},\mathbf{H},\mathbf{G}) \\
&\geq -\sum_{f=1}^{F}\sum_{t=1}^{T} u_f^{-1}v_t^{-1}[\boldsymbol{\sigma}_{\boldsymbol{\theta}}^2(\mathbf{z}_t)]_f^{-1}\mathrm{tr}\big(\mathbf{G}_{0f}^{-1}\boldsymbol{\Phi}_{0ft}\mathbf{X}_{ft}\boldsymbol{\Phi}_{0ft}^{\mathsf{H}}\big) \\
&\quad -\sum_{f=1}^{F}\sum_{t=1}^{T}\sum_{n=1}^{N}\sum_{k=1}^{K} w_{nkf}^{-1}h_{nkt}^{-1}\mathrm{tr}\big(\mathbf{G}_{nf}^{-1}\boldsymbol{\Psi}_{nftk}\mathbf{X}_{ft}\boldsymbol{\Psi}_{nftk}^{\mathsf{H}}\big) \\
&\quad -\sum_{f=1}^{F}\sum_{t=1}^{T}\sum_{n=0}^{N}\lambda_{nft}\mathrm{tr}\big(\mathbf{G}_{nf}\boldsymbol{\Omega}_{ft}^{-1}\big) - \sum_{f=1}^{F}\sum_{t=1}^{T}\log|\boldsymbol{\Omega}_{ft}| + \mathrm{const} \qquad (3.18)\\
&\triangleq \mathcal{L}_{\mathrm{FR}}^{2}(\mathbf{Z},\mathbf{U},\mathbf{V},\mathbf{W},\mathbf{H},\mathbf{G},\boldsymbol{\Omega},\boldsymbol{\Phi},\boldsymbol{\Psi}), \qquad (3.19)
\end{aligned}$$

where the equality conditions are Eq. (3.16), Eq. (3.17), and

$$\boldsymbol{\Psi}_{nftk} = \mathbf{Y}_{nftk}\mathbf{Y}_{ft}^{-1}. \qquad (3.20)$$

Note that $\mathbf{Y}_{ft} \triangleq \sum_{n=0}^{N}\mathbf{Y}_{nft}$ and $\mathbf{Y}_{nftk} \triangleq w_{nkf}h_{nkt}\mathbf{G}_{nf}$ $(n \geq 1)$. Since $\mathcal{L}_{\mathrm{FR}}^{1}$ is tighter than $\mathcal{L}_{\mathrm{FR}}^{2}$, it is better to use $\mathcal{L}_{\mathrm{FR}}^{1}$ for parameter estimation if possible. However, maximization of $\mathcal{L}_{\mathrm{FR}}^{1}$ with respect to $\mathbf{W}$ and $\mathbf{H}$ has no closed-form solution due to the existence of $\lambda_{nft}^{-1} = (\sum_k w_{nkf}h_{nkt})^{-1}$ $(n \geq 1)$ in the first term of Eq. (3.14). We thus use $\mathcal{L}_{\mathrm{FR}}^{1}$ for estimating $\mathbf{Z}$, $\mathbf{U}$, $\mathbf{V}$, and $\mathbf{G}$, and use $\mathcal{L}_{\mathrm{FR}}^{2}$ for $\mathbf{W}$ and $\mathbf{H}$.

Updating Speech Model

To update the latent variables $\mathbf{Z}$, we use the Metropolis sampling [130] or the backpropagation [131]. In the sampling, a proposal $\mathbf{z}_t^{\mathrm{new}} \sim \mathcal{N}(\mathbf{z}_t^{\mathrm{old}}, \xi\mathbf{I}_D)$ with a small number ξ is accepted as a next sample of $\mathbf{z}_t$ with probability $\beta_t \triangleq \min(1, \gamma_t)$, where γ_t is given by

$$\begin{aligned}
\log\gamma_t &= \mathcal{L}_{\mathrm{FR}}^{1}(\mathbf{z}_t^{\mathrm{new}},\mathbf{U},\mathbf{V},\mathbf{W},\mathbf{H},\mathbf{G},\boldsymbol{\Omega},\boldsymbol{\Phi}) + \log p(\mathbf{z}_t^{\mathrm{new}}) \\
&\quad - \mathcal{L}_{\mathrm{FR}}^{1}(\mathbf{z}_t^{\mathrm{old}},\mathbf{U},\mathbf{V},\mathbf{W},\mathbf{H},\mathbf{G},\boldsymbol{\Omega},\boldsymbol{\Phi}) - \log p(\mathbf{z}_t^{\mathrm{old}}) \qquad (3.21)\\
&= -\sum_{f=1}^{F}\left(\frac{1}{\lambda_{0ft}^{\mathrm{new}}} - \frac{1}{\lambda_{0ft}^{\mathrm{old}}}\right)\mathrm{tr}\big(\mathbf{G}_{0f}^{-1}\boldsymbol{\Phi}_{0ft}\mathbf{X}_{ft}\boldsymbol{\Phi}_{0ft}^{\mathsf{H}}\big) - \frac{1}{2}\sum_{d=1}^{D}\big((z_{td}^{\mathrm{new}})^2 - (z_{td}^{\mathrm{old}})^2\big) \\
&\quad -\sum_{f=1}^{F}\big(\lambda_{0ft}^{\mathrm{new}} - \lambda_{0ft}^{\mathrm{old}}\big)\,\mathrm{tr}\big(\mathbf{G}_{0f}\boldsymbol{\Omega}_{ft}^{-1}\big)\,, \qquad (3.22)
\end{aligned}$$

where $\lambda_{0ft}^{\mathrm{new}} \triangleq u_f v_t [\boldsymbol{\sigma}_{\boldsymbol{\theta}}^2(\mathbf{z}_t^{\mathrm{new}})]_f$ and $\lambda_{0ft}^{\mathrm{old}} \triangleq u_f v_t [\boldsymbol{\sigma}_{\boldsymbol{\theta}}^2(\mathbf{z}_t^{\mathrm{old}})]_f$. In the backpropagation, the lower bound $\mathcal{L}_{\mathrm{FR}}^1$ given by Eq. (3.15) is regarded as an objective function of $\mathbf{Z}$. It is maximized with respect to $\mathbf{z}_t$ by using a stochastic gradient ascent method. Both sampling and backpropagation algorithms update $\mathbf{Z}$ several times in one iteration. In practice, we update $\mathbf{Z}$ several times without updating $\mathbf{Y}_{ft}$ to reduce the computational cost of calculating $\mathbf{Y}_{ft}^{-1}$ in $\boldsymbol{\Phi}_{0ft}$ and $\boldsymbol{\Omega}_{ft}^{-1}$.

To derive the multiplicative updating (MU) rule of the scaling factors $\mathbf{U}$, we let the partial derivative of $\mathcal{L}_{\mathrm{FR}}^1$ given by Eq. (3.15) with respect to u_f equal to zero as follows:

$$\sum_{t=1}^{T} u_f^{-2} v_t^{-1} [\boldsymbol{\sigma}_{\boldsymbol{\theta}}^2(\mathbf{z}_t)]_f^{-1} \mathrm{tr}\big(\mathbf{G}_{0f}^{-1} \boldsymbol{\Phi}_{0ft} \mathbf{X}_{ft} \boldsymbol{\Phi}_{0ft}^{\mathsf{H}}\big) - \sum_{t=1}^{T} v_t [\boldsymbol{\sigma}_{\boldsymbol{\theta}}^2(\mathbf{z}_t)]_f \mathrm{tr}\big(\mathbf{G}_{0f} \boldsymbol{\Omega}_{ft}^{-1}\big) = 0. \tag{3.23}$$

Solving Eq. (3.23) for u_f, we have

$$u_f = \sqrt{\frac{\sum_{t=1}^{T} v_t^{-1} [\boldsymbol{\sigma}_{\boldsymbol{\theta}}^2(\mathbf{z}_t)]_f^{-1} \mathrm{tr}\left(\mathbf{G}_{0f}^{-1} \boldsymbol{\Phi}_{0ft} \mathbf{X}_{ft} \boldsymbol{\Phi}_{0ft}^{\mathsf{H}}\right)}{\sum_{t=1}^{T} v_t [\boldsymbol{\sigma}_{\boldsymbol{\theta}}^2(\mathbf{z}_t)]_f \mathrm{tr}\left(\mathbf{G}_{0f} \boldsymbol{\Omega}_{ft}^{-1}\right)}}. \tag{3.24}$$

Substituting $\boldsymbol{\Omega}_{ft} = \mathbf{Y}_{ft}$ and $\boldsymbol{\Phi}_{0ft} = \mathbf{Y}_{0ft} \mathbf{Y}_{ft}^{-1} = u_f v_t [\boldsymbol{\sigma}_{\boldsymbol{\theta}}^2(\mathbf{z}_t)]_f \mathbf{G}_{0f} \mathbf{Y}_{ft}^{-1}$ including the current estimate of u_f into Eq. (3.24), we have the MU rule of u_f given by

$$u_f \leftarrow u_f \sqrt{\frac{\sum_{t=1}^{T} v_t [\boldsymbol{\sigma}_{\boldsymbol{\theta}}^2(\mathbf{z}_t)]_f \mathrm{tr}\big(\mathbf{G}_{0f} \mathbf{Y}_{ft}^{-1} \mathbf{X}_{ft} \mathbf{Y}_{ft}^{-1}\big)}{\sum_{t=1}^{T} v_t [\boldsymbol{\sigma}_{\boldsymbol{\theta}}^2(\mathbf{z}_t)]_f \mathrm{tr}\big(\mathbf{G}_{0f} \mathbf{Y}_{ft}^{-1}\big)}}. \tag{3.25}$$

Similarly, the MU rule of the activations $\mathbf{V}$ can be obtained as follows:

$$v_t \leftarrow v_t \sqrt{\frac{\sum_{f=1}^{F} u_f [\boldsymbol{\sigma}_{\boldsymbol{\theta}}^2(\mathbf{z}_t)]_f \, \mathrm{tr}\big(\mathbf{G}_{0f} \mathbf{Y}_{ft}^{-1} \mathbf{X}_{ft} \mathbf{Y}_{ft}^{-1}\big)}{\sum_{f=1}^{F} u_f [\boldsymbol{\sigma}_{\boldsymbol{\theta}}^2(\mathbf{z}_t)]_f \, \mathrm{tr}\big(\mathbf{G}_{0f} \mathbf{Y}_{ft}^{-1}\big)}}. \tag{3.26}$$

Updating Noise Models

We derive the MU rules for $\mathbf{W}$ and $\mathbf{H}$ by almost the same way as $\mathbf{U}$ and $\mathbf{V}$. Letting the partial derivatives of $\mathcal{L}_{\mathrm{FR}}^2$ given by Eq. (3.19) with respect to w_{nkf} ($n \geq 1$) equal to zero, we have

$$\sum_{t=1}^{T} w_{nkf}^{-2} h_{nkt}^{-1} \mathrm{tr}\big(\mathbf{G}_{nf}^{-1} \boldsymbol{\Psi}_{nftk} \mathbf{X}_{ft} \boldsymbol{\Psi}_{nftk}^{\mathsf{H}}\big) - \sum_{t=1}^{T} h_{nkt} \mathrm{tr}\big(\mathbf{G}_{nf} \boldsymbol{\Omega}_{ft}^{-1}\big) = 0. \tag{3.27}$$

Solving Eq. (3.27) for w_{nkf}, we have

$$w_{nkf} = \sqrt{\frac{\sum_{t=1}^{T} h_{nkt}^{-1} \mathrm{tr}\left(\mathbf{G}_{nf}^{-1} \mathbf{\Psi}_{nftk} \mathbf{X}_{ft} \mathbf{\Psi}_{nftk}^{\mathsf{H}}\right)}{\sum_{t=1}^{T} h_{nkt} \mathrm{tr}\left(\mathbf{G}_{nf} \mathbf{\Omega}_{ft}^{-1}\right)}}. \tag{3.28}$$

Substituting $\mathbf{\Omega}_{ft} = \mathbf{Y}_{ft}$ and $\mathbf{\Psi}_{nftk} = \mathbf{Y}_{nftk}\mathbf{Y}_{ft}^{-1} = w_{nkf}h_{nkt}\mathbf{G}_{nf}\mathbf{Y}_{ft}^{-1}$ including the current estimate of w_{nkf} into Eq. (3.28), the closed-form MU rule of $\mathbf{W}$ is obtained as follows:

$$w_{nkf} \leftarrow w_{nkf} \sqrt{\frac{\sum_{t=1}^{T} h_{nkt} \, \mathrm{tr}\left(\mathbf{G}_{nf} \mathbf{Y}_{ft}^{-1} \mathbf{X}_{ft} \mathbf{Y}_{ft}^{-1}\right)}{\sum_{t=1}^{T} h_{nkt} \, \mathrm{tr}\left(\mathbf{G}_{nf} \mathbf{Y}_{ft}^{-1}\right)}}, \tag{3.29}$$

Similarly, the MU rule of $\mathbf{H}$ can be obtained as follows:

$$h_{nkt} \leftarrow h_{nkt} \sqrt{\frac{\sum_{f=1}^{F} w_{nkf} \, \mathrm{tr}\left(\mathbf{G}_{nf} \mathbf{Y}_{ft}^{-1} \mathbf{X}_{ft} \mathbf{Y}_{ft}^{-1}\right)}{\sum_{f=1}^{F} w_{nkf} \, \mathrm{tr}\left(\mathbf{G}_{nf} \mathbf{Y}_{ft}^{-1}\right)}}. \tag{3.30}$$

Updating Spatial Models

To derive the update rule of the spatial covariance matrices $\mathbf{G}$, we let the partial derivative of $\mathcal{L}_{\mathrm{FR}}^{1}$ with respect to $\mathbf{G}_{nf}$ equal to zero as follows:

$$\sum_{t=1}^{T} \lambda_{nft}^{-1} \mathbf{G}_{nf}^{-1} \mathbf{\Phi}_{nft} \mathbf{X}_{ft} \mathbf{\Phi}_{nft}^{\mathsf{H}} \mathbf{G}_{nf}^{-1} - \sum_{t=1}^{T} \lambda_{nft} \mathbf{\Omega}_{ft}^{-1} = \mathbf{0}_{M \times M}, \tag{3.31}$$

where $\mathbf{0}_{M \times M}$ is the all-zero matrix of size $M \times M$. Eq. (3.31) can be rewritten as follows:

$$\mathbf{G}_{nf} \left(\sum_{t=1}^{T} \lambda_{nft} \mathbf{\Omega}_{ft}^{-1} \right) \mathbf{G}_{nf} = \sum_{t=1}^{T} \lambda_{nft}^{-1} \mathbf{\Phi}_{nft} \mathbf{X}_{ft} \mathbf{\Phi}_{nft}^{\mathsf{H}}. \tag{3.32}$$

Solving Eq. (3.32) as in [49,51], we have the closed-form update rule of $\mathbf{G}_{nf}$ as follows:

$$\mathbf{G}_{nf} \leftarrow \left(\sum_{t=1}^{T} \lambda_{nft} \mathbf{\Omega}_{ft}^{-1} \right)^{-1} \# \left(\sum_{t=1}^{T} \lambda_{nft}^{-1} \mathbf{\Phi}_{nft} \mathbf{X}_{ft} \mathbf{\Phi}_{nft}^{\mathsf{H}} \right). \tag{3.33}$$

where $\mathbf{A}\#\mathbf{B}$ indicates the geometric mean of two positive definite matrices $\mathbf{A}$ and $\mathbf{B}$ [132,133] as follows:

$$\mathbf{A}\#\mathbf{B} = \mathbf{A}^{\frac{1}{2}} \left(\mathbf{A}^{-\frac{1}{2}} \mathbf{B} \mathbf{A}^{-\frac{1}{2}} \right)^{\frac{1}{2}} \mathbf{A}^{\frac{1}{2}} = \mathbf{A}^{\frac{1}{2}} \left(\mathbf{A}^{\frac{1}{2}} \mathbf{A}^{-1} \mathbf{B} \mathbf{A}^{-\frac{1}{2}} \right)^{\frac{1}{2}} \mathbf{A}^{\frac{1}{2}} \tag{3.34}$$

$$= \mathbf{A}^{\frac{1}{2}} \left(\left(\mathbf{A}^{\frac{1}{2}} (\mathbf{A}^{-1}\mathbf{B})^{\frac{1}{2}} \mathbf{A}^{-\frac{1}{2}} \right)^2 \right)^{\frac{1}{2}} \mathbf{A}^{\frac{1}{2}} = \mathbf{A}(\mathbf{A}^{-1}\mathbf{B})^{\frac{1}{2}}. \tag{3.35}$$

Let $\mathbf{C}$ be a square matrix whose eigenvalues are positive, and let the eigenvalue decomposition of $\mathbf{C}$ be $\mathbf{C} = \mathbf{P}\mathbf{\Lambda}\mathbf{P}^{-1}$. $\mathbf{C}^{\frac{1}{2}}$ is defined as $\mathbf{C}^{\frac{1}{2}} \triangleq \mathbf{P}\mathbf{\Lambda}^{\frac{1}{2}}\mathbf{P}^{-1}$, where $\mathbf{\Lambda}^{\frac{1}{2}}$ is the diagonal matrix whose diagonal elements are the square roots of those of $\mathbf{\Lambda}$. While the product of two positive definite matrices $\mathbf{AB}$ is not a Hermitian matrix, the eigenvalues of $\mathbf{AB}$ are positive. To prove this, we first prove that, for matrices $\mathbf{S} \in \mathbb{C}^{m \times n}$ and $\mathbf{T} \in \mathbb{C}^{n \times m}$, $\mathbf{ST}$ and $\mathbf{TS}$ have the same eigenvalues. Assuming that λ and $\mathbf{u}$ are an eigenvalue and eigenvector of $\mathbf{ST}$, respectively, the following equations hold:

$$\mathbf{STu} = \lambda \mathbf{u}, \tag{3.36}$$

$$\mathbf{TSTu} = \lambda \mathbf{Tu}. \tag{3.37}$$

This indicates that λ and $\mathbf{Tu}$ are an eigenvalue and eigenvector of $\mathbf{TS}$, respectively. Thus, $\mathbf{AB} = \mathbf{A}^{\frac{1}{2}} \left(\mathbf{A}^{\frac{1}{2}} \mathbf{B} \right)$ has the same eigenvalues as $\left(\mathbf{A}^{\frac{1}{2}} \mathbf{B} \right) \mathbf{A}^{\frac{1}{2}}$, which is a positive definite matrix because $\forall \mathbf{x}$, $\mathbf{x}^{\mathsf{H}} \mathbf{A}^{\frac{1}{2}} \mathbf{B} \mathbf{A}^{\frac{1}{2}} \mathbf{x} = \left(\mathbf{A}^{\frac{1}{2}} \mathbf{x} \right)^{\mathsf{H}} \mathbf{B} \left(\mathbf{A}^{\frac{1}{2}} \mathbf{x} \right) > 0$ holds.

Substituting $\mathbf{\Omega}_{ft} = \mathbf{Y}_{ft}$ and $\mathbf{\Phi}_{nft} = \mathbf{Y}_{nft}\mathbf{Y}_{ft}^{-1} = \lambda_{nft}\mathbf{G}_{nf}\mathbf{Y}_{ft}^{-1}$ including the current estimate of $\mathbf{G}_{nf}$ into Eq. (3.33), we have

$$\mathbf{G}_{nf} \leftarrow \left(\sum_{t=1}^{T} \lambda_{nft} \mathbf{Y}_{ft}^{-1} \right)^{-1} \# \left(\mathbf{G}_{nf} \left(\sum_{t=1}^{T} \lambda_{nft} \mathbf{Y}_{ft}^{-1} \mathbf{X}_{ft} \mathbf{Y}_{ft}^{-1} \right) \mathbf{G}_{nf} \right). \tag{3.38}$$

Normalizing Parameters

To meet the normalization constraints given by Eq. (3.9), Eq. (3.10), and Eq. (3.11), we adjust the scales of $\mathbf{U}$, $\mathbf{V}$, $\mathbf{W}$, $\mathbf{H}$, and $\mathbf{G}$ in each iteration as follows:

$$\mu_{nf} \triangleq \mathrm{tr}(\mathbf{G}_{nf}), \quad \begin{cases} \mathbf{G}_{nf} \leftarrow \mu_{nf}^{-1} \mathbf{G}_{nf}, \\ u_f \leftarrow \mu_{0f} u_f, \\ w_{nkf} \leftarrow \mu_{nf} w_{nkf} \ (n \geq 1), \end{cases} \tag{3.39}$$

$$\nu_0 \triangleq \sum_{f=1}^{F} u_f, \quad \begin{cases} u_f \leftarrow \nu_0^{-1} u_f, \\ v_t \leftarrow \nu_0 v_t, \end{cases} \tag{3.40}$$

$$\nu_{nk} \triangleq \sum_{f=1}^{F} w_{nkf}, \quad \begin{cases} w_{nkf} \leftarrow \nu_{nk}^{-1} w_{nkf}, \\ h_{nkt} \leftarrow \nu_{nk} h_{nkt}. \end{cases} \tag{3.41}$$

Algorithm 1 Speech enhancement based on MNMF-DSP.

for iteration = 1 to MaxIteration **do**
  Update $\mathbf{U}$, $\mathbf{V}$, $\mathbf{W}$, $\mathbf{H}$, and $\mathbf{G}$ by Eqs. (3.25), (3.26), (3.29), (3.30), and (3.38)
  Compute $\mathbf{\Omega}$ and $\{\mathbf{\Phi}_{0ft}\}_{f,t=1}^{F,T}$ by Eqs. (3.16) and (3.17)
  if Sampling **then**
    for Z_iteration = 1 to Z_MaxIteration **do**
      for $t = 1$ to T **do**
        Sample $\mathbf{z}_t^{\mathrm{new}}$ from $\mathcal{N}_{\mathbb{C}}(\mathbf{z}_t, \xi \mathbf{I}_D)$
        Compute γ_t by Eq. (3.22)
        Sample q from $\mathrm{Uniform}(0, 1)$
        if $\gamma_t > q$ **then** $\mathbf{z}_t \leftarrow \mathbf{z}_t^{\mathrm{new}}$
      end for
    end for
  end if
  if Backpropagation **then**
    for Z_iteration = 1 to Z_MaxIteration **do**
      Compute $\mathcal{L}_{\mathrm{FR}}^1$ by Eq. (3.14)
      Update $\mathbf{Z}$ by Adam with $\mathcal{L}_{\mathrm{FR}}^1$
    end for
  end if
  Normalize parameters by Eqs. (3.39), (3.40), and (3.41)
end for
Compute $\mathbf{x}_{0ft}^{\mathrm{FR}}$ by Eq. (1.7)

3.3 ILRMA with a Deep Speech Prior (ILRMA-DSP)

We formulate a unified probabilistic generative model by integrating the DNN-based speech model, NMF-based noise model, and the rank-1 spatial model described in Section 1.3.2 and then derive an update rule based on an MM algorithm.

3.3.1 Formulation

We assume the number of microphones M is equal to $N + 1$ to derive an efficient update rule. Now the mixing matrix $\mathbf{A}_f \in \mathbb{C}^{M \times N+1}$ becomes a non-singular square matrix, and the estimation value of the source spectrum, $\tilde{\mathbf{s}}_{ft} \triangleq [\tilde{s}_{0ft}, \ldots, \tilde{s}_{Nft}]^{\mathsf{T}}$, is given as follows:

$$\tilde{\mathbf{s}}_{ft} = \mathbf{D}_f \mathbf{x}_{ft}, \tag{3.42}$$

where $\mathbf{D}_f = \mathbf{A}_f^{-1} = [\mathbf{d}_{1f}, \ldots, \mathbf{d}_{Nf}]^{\mathsf{H}} \in \mathbb{C}^{N \times M}$ is a demixing matrix. When $\mathbf{G}_{nf}$ is a rank-1 matrix given by $\mathbf{G}_{nf} = \mathbf{a}_{nf}\mathbf{a}_{nf}^{\mathsf{H}}$, $\mathbf{Y}_{ft}$ is given as follows:

$$\begin{aligned}\mathbf{Y}_{ft} &= \sum_{n=0}^{N} \lambda_{nft}\mathbf{a}_{nf}\mathbf{a}_{nf}^{\mathsf{H}} \\ &= \mathbf{A}_f\mathbf{\Lambda}_{ft}\mathbf{A}_f^{\mathsf{H}} = \mathbf{D}_f^{-1}\mathbf{\Lambda}_{ft}\mathbf{D}_f^{-\mathsf{H}},\end{aligned} \tag{3.43}$$

where $\mathbf{\Lambda}_{ft} \triangleq \mathrm{Diag}(\lambda_{0ft}, \ldots, \lambda_{nft})$ is a diagonal matrix. Substituting Eq. (3.42) and Eq. (3.43) into Eq. (3.5), we get

$$\begin{aligned}&\log p(\mathbf{X}|\mathbf{Z}, \mathbf{U}, \mathbf{V}, \mathbf{W}, \mathbf{H}, \mathbf{D}) \\ &= -\sum_{f=1}^{F}\sum_{t=1}^{T} \mathrm{tr}\left(\tilde{\mathbf{s}}_{ft}^{\mathsf{H}}\mathbf{D}_f^{-\mathsf{H}}\left(\mathbf{D}_f^{\mathsf{H}}\mathbf{\Lambda}_{ft}^{-1}\mathbf{D}_f\right)\mathbf{D}_f^{-1}\tilde{\mathbf{s}}_{ft}\right) - \sum_{f=1}^{F}\sum_{t=1}^{T}\log\left|\mathbf{D}_f^{-1}\mathbf{\Lambda}_{ft}\mathbf{D}_f^{-\mathsf{H}}\right| + \mathrm{const} \qquad (3.44)\\ &= -\sum_{f=1}^{F}\sum_{t=1}^{T} \mathrm{tr}\left(\tilde{\mathbf{s}}_{ft}^{\mathsf{H}}\mathbf{\Lambda}_{ft}^{-1}\tilde{\mathbf{s}}_{ft}\right) - \sum_{f=1}^{F}\sum_{t=1}^{T}\log|\mathbf{\Lambda}_{ft}| + T\sum_{f=1}^{F}\log|\mathbf{D}_f\mathbf{D}_f^{\mathsf{H}}| + \mathrm{const} \qquad (3.45)\\ &= -\sum_{f=1}^{F}\sum_{t=1}^{T}\sum_{n=0}^{N}\left(\frac{|\tilde{s}_{nft}|^2}{\lambda_{nft}} + \log\lambda_{nft}\right) + T\sum_{f=1}^{F}\log|\mathbf{D}_f\mathbf{D}_f^{\mathsf{H}}| + \mathrm{const.} \qquad (3.46)\end{aligned}$$

Our goal is to estimate the demixing matrices $\mathbf{D}$ instead of the mixing matrices $\mathbf{A}$ and to estimate $\mathbf{Z}$, $\mathbf{U}$, $\mathbf{V}$, $\mathbf{W}$, and $\mathbf{H}$ such that the log-likelihood $\log p(\mathbf{X}|\mathbf{Z}, \mathbf{U}, \mathbf{V}, \mathbf{W}, \mathbf{H}, \mathbf{D})$ given by Eq. (3.46) is maximized. To avoid the scale ambiguity of the parameters, we put the normalization constraints on $\mathbf{U}$ and $\mathbf{W}$ given by Eq. (3.9) and Eq. (3.10) and that on $\mathbf{D}$ given by

$$\mathrm{tr}\left(\mathbf{d}_{nf}\mathbf{d}_{nf}^{\mathsf{H}}\right) = \mathbf{d}_{nf}^{\mathsf{H}}\mathbf{d}_{nf} = 1. \tag{3.47}$$

3.3.2 Optimization

We aim to estimate the parameters $\mathbf{Z}$, $\mathbf{U}$, $\mathbf{V}$, $\mathbf{W}$, $\mathbf{H}$, and $\mathbf{D}$ that maximize $\log p(\mathbf{X}|\mathbf{Z}, \mathbf{U}, \mathbf{V}, \mathbf{W}, \mathbf{H}, \mathbf{D})$ given by Eq. (3.46) by using an MM algorithm as in ILRMA [28] and obtain the enhanced speech image $\mathbf{x}_{0ft}^{\mathrm{R1}} \in \mathbb{C}^M$ using a linear demixing filter given by Eq. (1.9).

Updating Speech Model

The latent variables $\mathbf{Z}$ are updated with Metropolis sampling or backpropagation as in the full-rank model (Section 3.2.2). In the sampling, instead of Eq. (3.22), γ_t is given by

$$\log \gamma_t = \sum_{f=1}^{F} \left(\frac{|\tilde{s}_{0ft}|^2}{\lambda_{0ft}^{\mathrm{old}}} - \frac{|\tilde{s}_{0ft}|^2}{\lambda_{0ft}^{\mathrm{new}}} + \log \frac{\lambda_{0ft}^{\mathrm{old}}}{\lambda_{0ft}^{\mathrm{new}}} \right) - \frac{1}{2} \sum_{d=1}^{D} \left((z_{td}^{\mathrm{new}})^2 - (z_{td}^{\mathrm{old}})^2 \right). \tag{3.48}$$

In the backpropagation, the likelihood given by Eq. (3.46) is regarded as a negative cost function.

The update rules of $\mathbf{U}$ and $\mathbf{V}$ can be obtained directly by letting the partial derivatives of $\log p(\mathbf{X}|\mathbf{Z}, \mathbf{U}, \mathbf{V}, \mathbf{W}, \mathbf{H}, \mathbf{D})$ equal to zero as follows:

$$u_f \leftarrow \frac{1}{T} \sum_{t=1}^{T} \frac{|\tilde{s}_{0ft}|^2}{v_t [\boldsymbol{\sigma}_{\boldsymbol{\theta}}^2(\mathbf{z}_t)]_f}, \tag{3.49}$$

$$v_t \leftarrow \frac{1}{F} \sum_{f=1}^{F} \frac{|\tilde{s}_{0ft}|^2}{u_f [\boldsymbol{\sigma}_{\boldsymbol{\theta}}^2(\mathbf{z}_t)]_f}. \tag{3.50}$$

Updating Noise Models

The closed-form MU rules of $\mathbf{W}$ and $\mathbf{H}$ are obtained in the same way as [8] as follows:

$$w_{nkf} \leftarrow w_{nkf} \sqrt{\frac{\sum_{t=1}^{T} h_{nkt} |\tilde{s}_{nft}|^2 \lambda_{nft}^{-2}}{\sum_{t=1}^{T} h_{nkt} \lambda_{nft}^{-1}}}, \tag{3.51}$$

$$h_{nkt} \leftarrow h_{nkt} \sqrt{\frac{\sum_{f=1}^{F} w_{nkf} |\tilde{s}_{nft}|^2 \lambda_{nft}^{-2}}{\sum_{f=1}^{F} w_{nkf} \lambda_{nft}^{-1}}}. \tag{3.52}$$

Updating Spatial Models

The update rule of $\mathbf{D}$ is obtained in the same way as [28, 70] as follows:

$$\boldsymbol{\Upsilon}_{nf} \triangleq \frac{1}{T} \sum_{t=1}^{T} \frac{\mathbf{X}_{ft}}{\lambda_{nft}}, \tag{3.53}$$

$$\mathbf{d}_{nf} \leftarrow (\mathbf{D}_f \boldsymbol{\Upsilon}_{nf})^{-1} \mathbf{e}_n, \tag{3.54}$$

Algorithm 2 Speech enhancement based on ILRMA-DSP.

for iteration = 1 to MaxIteration **do**
  Update $\mathbf{U}$, $\mathbf{V}$, $\mathbf{W}$, and $\mathbf{H}$ by Eqs. (3.49), (3.50), (3.51), and (3.52)
  if Sampling **then**
    for Z_iteration = 1 to Z_MaxIteration **do**
      for $t = 1$ to T **do**
        Sample $\mathbf{z}_t^{\text{new}}$ from $\mathcal{N}_{\mathbb{C}}(\mathbf{z}_t, \xi \mathbf{I}_D)$
        Compute γ_t by Eq. (3.48)
        Sample q from $\text{Uniform}(0, 1)$
        if $\gamma_t > q$ **then** $\mathbf{z}_t \leftarrow \mathbf{z}_t^{\text{new}}$
      end for
    end for
  end if
  if Backpropagation **then**
    for Z_iteration = 1 to Z_MaxIteration **do**
      Compute the log likelihood by Eq. (3.46)
      Update $\mathbf{Z}$ by Adam with the log likelihood
    end for
  end if
  Update $\mathbf{D}$ by Eq. (3.54)
  Normalize parameters by Eqs. (3.40), (3.41), and (3.56)
end for
Compute $\mathbf{x}_{0ft}^{\text{R1}}$ by Eq. (1.9)

$$\mathbf{d}_{nf} \leftarrow \left(\mathbf{d}_{nf}^{\mathsf{H}} \boldsymbol{\Upsilon}_{nf} \mathbf{d}_{nf}\right)^{-\frac{1}{2}} \mathbf{d}_{nf}, \tag{3.55}$$

where $\mathbf{e}_n \triangleq [0, \ldots, 1, \ldots, 0]^{\mathsf{T}}$ indicates a unit vector with the n-th element equal to 1.

Normalizing Parameters

To meet the normalization constraints given by Eq. (3.9), Eq. (3.10), and Eq. (3.47), we normalize $\mathbf{D}$ as follows:

$$\mu_{nf} \triangleq \mathbf{d}_{nf}^{\mathsf{H}} \mathbf{d}_{nf}, \quad \begin{cases} \mathbf{d}_{nf} \leftarrow \mu_{nf}^{-\frac{1}{2}} \mathbf{d}_{nf}, \\ u_f \leftarrow \mu_{0f}^{-1} u_f, \\ w_{nkf} \leftarrow \mu_{nf}^{-1} w_{nkf} \ (n \geq 1), \end{cases} \tag{3.56}$$

We then normalize $\mathbf{U}$ and $\mathbf{W}$ by using Eq. (3.40) and Eq. (3.41).

3.4 Initialization

It is crucial to appropriately initialize the scaling factors $\mathbf{U}$, the speech activations $\mathbf{V}$, the speech latent variables $\mathbf{Z}$, the basis spectra $\mathbf{W}$, the noise activations $\mathbf{H}$, and the SCMs $\mathbf{G}$ or the demixing matrices $\mathbf{D}$. We use the inference model of the VAE specified by $\boldsymbol{\phi}$ for initializing $\mathbf{Z}$ as $\mathbf{z}_t \leftarrow \boldsymbol{\mu}_{\boldsymbol{\phi}}(\mathbf{x}_t)$. $\mathbf{U}$ and $\mathbf{V}$ are initialized as $\mathbf{u} = \frac{1}{F}\mathbf{1}_F$ and $\mathbf{v} = \mathbf{1}_T$.

Considering Eq. (3.10), the initial values of $\mathbf{W}$ are sampled from a Dirichlet distribution as follows:

$$\mathbf{w}_{nk} \sim \mathrm{Dirichlet}(\alpha_0 \mathbf{1}_F)\,, \tag{3.57}$$

where $\mathbb{E}_{\mathrm{init}}[w_{nkf}] = \frac{1}{F}$ and α_0 is a concentration parameter ($\alpha_0 = 2$ in our experiments). Considering Eq. (3.10), Eq. (3.11), and the scale of the observed PSDs, the initial values of $\mathbf{H}$ are sampled from gamma distributions as follows:

$$h_{nkt} \sim \mathrm{Gamma}\left(\alpha_0, \frac{\alpha_0}{\mathbb{E}_{\mathrm{emp}}[|x|^2]}\frac{NK}{FM}\right), \tag{3.58}$$

where $\mathbb{E}_{\mathrm{init}}[h_{nkt}] = \frac{FM}{NK}\mathbb{E}_{\mathrm{emp}}[|x|^2]$ and $\mathbb{E}_{\mathrm{emp}}[|x|^2]$ indicates the empirical mean of the observed PSDs given by

$$\mathbb{E}_{\mathrm{emp}}\left[|x|^2\right] = \frac{1}{FTM}\sum_{f=1}^{F}\sum_{t=1}^{T}\sum_{m=1}^{M}|x_{ftm}|^2. \tag{3.59}$$

Since the initialization of $\mathbf{G}$ or $\mathbf{D}$ is considered to have a strong impact on the performance of speech enhancement, we propose and compare several initialization methods.

3.4.1 MNMF-DSP

$\mathbf{G}$ can be initialized without using the observed data $\mathbf{X}$. The most naive way of initialization is to set $\mathbf{G}_{nf}$ to the identity matrix as follows:

$$\mathbf{G}_{nf} \leftarrow \frac{1}{M}\mathbf{I}_M. \tag{3.60}$$

Alternatively, $\mathbf{G}$ can be initialized in an adaptive manner by using the observed data $\mathbf{X}$. Assuming that the target speech is predominant in $\mathbf{X}$, one may set the

speech SCM $\mathbf{G}_{0f}$ to the average of the observed SCMs and the noise SCMs to the identity matrix as follows:

$$\begin{cases} \mathbf{G}_{0f} \leftarrow \dfrac{\sum_{t=1}^{T} \mathbf{X}_{ft}}{\sum_{t=1}^{T} \mathrm{tr}(\mathbf{X}_{ft})}, \\ \mathbf{G}_{nf} \leftarrow \dfrac{1}{M}\mathbf{I}_M \ \ (n \geq 1). \end{cases} \tag{3.61}$$

A more sophisticated way of initialization is to use a fast speech enhancement method based on a complex Gaussian mixture model (cGMM) [89] that classifies each time-frequency bin into speech or noise. Here, we initialize the cGMM with Eq. (3.61). Using the estimated posterior probability ω_{ft} that the bin at frequency f and time t was generated from the speech, we have

$$\begin{cases} \mathbf{G}_{0f} \leftarrow \dfrac{\sum_{t=1}^{T} \omega_{ft}\mathbf{X}_{ft}}{\sum_{t=1}^{T} \omega_{ft}\mathrm{tr}(\mathbf{X}_{ft})}, \\ \mathbf{G}_{nf} \leftarrow \dfrac{\sum_{t=1}^{T} (1-\omega_{ft})\mathbf{X}_{ft}}{\sum_{t=1}^{T} (1-\omega_{ft})\mathrm{tr}(\mathbf{X}_{ft})} \ \ (n \geq 1). \end{cases} \tag{3.62}$$

3.4.2 ILRMA-DSP

In the determined condition of the rank-1 model, $\mathbf{D}$ cannot be initialized in a way corresponding to Eq. (3.60) because the identity matrix is a full-rank matrix. The most naive way of initialization is to set $\mathbf{D}_f$ to the identity matrix as follows:

$$\mathbf{D}_f \leftarrow \mathbf{I}_{N+1}, \ \text{ i.e., } \ \mathbf{d}_{nf} \leftarrow \mathbf{e}_{n+1}. \tag{3.63}$$

The demixing matrices $\mathbf{D}$ can alternatively be initialized in an adaptive manner by using the observed data $\mathbf{X}$. If the mixing matrix $\mathbf{A}_f = [\mathbf{a}_{0f}, \mathbf{a}_{1f}, \ldots, \mathbf{a}_{Nf}]$ is given, $\mathbf{D}_f$ is given by

$$\mathbf{D}_f \leftarrow \mathbf{A}_f^{-1}, \tag{3.64}$$

where $\mathbf{A}_f$ can be estimated from the full-rank SCMs $\mathbf{G}$. Using $\mathbf{G}_{0f}$ in Eq. (3.61) and $\{\mathbf{G}_{nf}\}_{n=1}^{N}$ in Eq. (3.60), we have

$$\begin{cases} \mathbf{a}_{0f} = \mathcal{PE}\left(\sum_{t=1}^{T} \mathbf{X}_{ft}\right), \\ \mathbf{a}_{nf} = \mathbf{e}_{n+1} \ \ (n \geq 1), \end{cases} \tag{3.65}$$

where $\mathcal{PE}(\cdot)$ indicates a normalized eigenvector that corresponds to the largest eigenvalue of a matrix. Alternatively, using Eq. (3.62), we have

$$\begin{cases} \mathbf{a}_{0f} = \mathcal{PE}\left(\sum_{t=1}^{T} \omega_{ft}\mathbf{X}_{ft}\right), \\ \mathbf{a}_{nf} = \mathcal{PE}\left(\sum_{t=1}^{T}(1-\omega_{ft})\mathbf{X}_{ft}\right) \quad (n \geq 1). \end{cases} \tag{3.66}$$

3.4.3 Pretraining of Deep Speech Prior

The nonlinear mapping function $\boldsymbol{\sigma}_{\boldsymbol{\theta}}^2(\cdot)$ given by Eq. (3.1) is optimized in the framework of a VAE. Suppose that we have training data $\tilde{\mathbf{X}} \triangleq \{\tilde{\mathbf{x}}_i\}_{i=1}^{I}$, where I is the number of frames and $\tilde{\mathbf{x}}_i \in \mathbb{C}^F$ is a complex spectrum of clean speech. Let $\tilde{\mathbf{Z}} \triangleq \{\tilde{\mathbf{z}}_i\}_{i=1}^{I}$ be the corresponding latent variables. We formulate the hierarchical generative process of $\tilde{\mathbf{X}}$ as follows:

$$\tilde{\mathbf{z}}_i \sim \mathcal{N}(\mathbf{0}_D, \mathbf{I}_D), \tag{3.67}$$

$$\tilde{\mathbf{x}}_i \sim \mathcal{N}_{\mathbb{C}}(\mathbf{0}_F, \mathrm{Diag}(\boldsymbol{\sigma}_{\boldsymbol{\theta}}^2(\tilde{\mathbf{z}}_i))), \tag{3.68}$$

where $\mathrm{Diag}(\cdot)$ indicates a diagonal matrix.

Our goal is to estimate $\boldsymbol{\theta}$ such that the likelihood $p(\tilde{\mathbf{X}}|\boldsymbol{\theta})$ is maximized. Since $\log p(\tilde{\mathbf{X}}|\boldsymbol{\theta})$ is analytically intractable and is hard to directly maximize, we derive a lower bound $\mathcal{L}_{\mathrm{VAE}}(\boldsymbol{\theta}, \boldsymbol{\phi})$ of $\log p(\tilde{\mathbf{X}}|\boldsymbol{\theta})$ by introducing a variational posterior distribution $q_{\boldsymbol{\phi}}(\tilde{\mathbf{z}}_i|\tilde{\mathbf{x}}_i)$ with parameters $\boldsymbol{\phi}$ as follows:

$$\log p(\tilde{\mathbf{X}}|\boldsymbol{\theta}) = \sum_{i=1}^{I} \log \int p_{\boldsymbol{\theta}}(\tilde{\mathbf{x}}_i|\tilde{\mathbf{z}}_i)p(\tilde{\mathbf{z}}_i)d\tilde{\mathbf{z}}_i \tag{3.69}$$

$$= \sum_{i=1}^{I} \log \int \frac{q_{\boldsymbol{\phi}}(\tilde{\mathbf{z}}_i|\tilde{\mathbf{x}}_i)}{q_{\boldsymbol{\phi}}(\tilde{\mathbf{z}}_i|\tilde{\mathbf{x}}_i)} p_{\boldsymbol{\theta}}(\tilde{\mathbf{x}}_i|\tilde{\mathbf{z}}_i)p(\tilde{\mathbf{z}}_i)d\tilde{\mathbf{z}}_i \tag{3.70}$$

$$\geq \sum_{i=1}^{I} \int q_{\boldsymbol{\phi}}(\tilde{\mathbf{z}}_i|\tilde{\mathbf{x}}_i) \log \frac{p_{\boldsymbol{\theta}}(\tilde{\mathbf{x}}_i|\tilde{\mathbf{z}}_i)p(\tilde{\mathbf{z}}_i)}{q_{\boldsymbol{\phi}}(\tilde{\mathbf{z}}_i|\tilde{\mathbf{x}}_i)} d\tilde{\mathbf{z}}_i \tag{3.71}$$

$$= \sum_{i=1}^{I} \left(\mathbb{E}_{q_{\boldsymbol{\phi}}}[\log p_{\boldsymbol{\theta}}(\tilde{\mathbf{x}}_i|\tilde{\mathbf{z}}_i)] - \mathrm{KL}(q_{\boldsymbol{\phi}}(\tilde{\mathbf{z}}_i|\tilde{\mathbf{x}}_i)\|p(\tilde{\mathbf{z}}_i))\right) \tag{3.72}$$

$$\triangleq \mathcal{L}_{\mathrm{VAE}}(\boldsymbol{\theta}, \boldsymbol{\phi}), \tag{3.73}$$

where $\mathrm{KL}(q\|p)$ indicates the Kullback-Leibler (KL) divergence between two

probability distributions q and p. Our goal is to maximize $\mathcal{L}_{\mathrm{VAE}}(\boldsymbol{\theta}, \boldsymbol{\phi})$ with respect to $\boldsymbol{\theta}$ and $\boldsymbol{\phi}$.

For mathematical convenience, $q_{\boldsymbol{\phi}}(\tilde{\mathbf{z}}_i|\tilde{\mathbf{x}}_i)$ is set to a Gaussian distribution as follows:

$$q_{\boldsymbol{\phi}}(\tilde{\mathbf{z}}_i|\tilde{\mathbf{x}}_i) = \mathcal{N}(\tilde{\mathbf{z}}_i|\boldsymbol{\mu}_{\boldsymbol{\phi}}(\tilde{\mathbf{x}}_i), \mathrm{Diag}(\boldsymbol{\sigma}^2_{\boldsymbol{\phi}}(\tilde{\mathbf{x}}_i))), \tag{3.74}$$

where $\boldsymbol{\mu}_{\boldsymbol{\phi}}(\cdot)$ and $\boldsymbol{\sigma}^2_{\boldsymbol{\phi}}(\cdot)$ are the D-dimensional output vectors of a DNN with parameters $\boldsymbol{\phi}$. The first term of Eq. (3.73) is approximated via Monte Carlo integration as follows:

$$\mathbb{E}_{q_{\boldsymbol{\phi}}}[\log p_{\boldsymbol{\theta}}(\tilde{\mathbf{x}}_i|\tilde{\mathbf{z}}_i)] \approx \frac{1}{L}\sum_{l=1}^{L} \log p_{\boldsymbol{\theta}}(\tilde{\mathbf{x}}_i|\tilde{\mathbf{z}}_i^{(l)}), \tag{3.75}$$

where L is the number of samples and $\tilde{\mathbf{z}}_i^{(l)}$ is obtained by using the reparametrization trick [134] as follows:

$$\tilde{\boldsymbol{\epsilon}}_i^{(l)} \sim \mathcal{N}(\mathbf{0}_D, \mathbf{I}_D), \tag{3.76}$$

$$\tilde{\mathbf{z}}_i^{(l)} = \boldsymbol{\mu}_{\boldsymbol{\phi}}(\tilde{\mathbf{x}}_i) + \tilde{\boldsymbol{\epsilon}}_i^{(l)} \odot \sqrt{\boldsymbol{\sigma}^2_{\boldsymbol{\phi}}(\tilde{\mathbf{x}}_i)}, \tag{3.77}$$

where $\odot$ indicates the Hadamard product. The second term of Eq. (3.73) can be analytically calculated as follows:

$$\mathrm{KL}(q_{\boldsymbol{\phi}}(\tilde{\mathbf{z}}_i|\tilde{\mathbf{x}}_i)\|p(\tilde{\mathbf{z}}_i)) = \frac{1}{2}\sum_{d=1}^{D}\left([\boldsymbol{\mu}_{\phi}(\tilde{\mathbf{x}}_i)]_d^2 + [\boldsymbol{\sigma}^2_{\boldsymbol{\phi}}(\tilde{\mathbf{x}}_i)]_d - \log[\boldsymbol{\sigma}^2_{\boldsymbol{\phi}}(\tilde{\mathbf{x}}_i)]_d - 1\right). \tag{3.78}$$

The lower bound $\mathcal{L}_{\mathrm{VAE}}(\boldsymbol{\theta}, \boldsymbol{\phi})$ given by Eq. (3.73) can be approximately calculated by using Eq. (3.75), Eq. (3.76), Eq. (3.77), and Eq. (3.78). The parameters $\boldsymbol{\theta}$ and $\boldsymbol{\phi}$ of the two DNNs are jointly optimized by using a stochastic gradient method such that $\mathcal{L}_{\mathrm{VAE}}(\boldsymbol{\theta}, \boldsymbol{\phi})$ is maximized.

The generation parameters $\boldsymbol{\theta}$ are used for formulating the generative model of $\mathbf{X}$ described in Section 3.1.1. The inference parameters $\boldsymbol{\phi}$ are used for initializing $\mathbf{Z}$, i.e., $\mathbf{z}_t \leftarrow \boldsymbol{\mu}_{\boldsymbol{\phi}}(\mathbf{x}_t)$ as described in Section 3.4, where $\mathbf{x}_t$ is any complex spectrum whose PSDs are the same as the average PSDs of noisy speech over all channels at frame t.

3.5 Evaluation

This section reports experiments conducted for investigating the performance of our semi-blind speech enhancement methods based on the MNMF-DSP or ILRMA-DSP with different configurations. First, we investigate the impacts of the model complexities (i.e., the number of noise sources N and the number of noise bases K) and verify the effectiveness of the low-rank noise model. We then evaluate the two methods used for optimizing the latent variables $\mathbf{Z}$ (i.e., Metropolis sampling and backpropagation methods described in Section 3.2.2 and Section 3.3.2) and the three methods used for initializing the spatial parameters $\mathbf{G}$ or $\mathbf{D}$ (i.e., identity-, observation-, and cGMM-based methods described in Section 3.4). Finally, we compare our method with the state-of-the-art blind, semi-blind, and supervised methods.

3.5.1 Configurations

Test Data

The simulated data sampled at 16 kHz in the evaluation dataset of CHiME3 [135] were used for evaluation. It contains 1320 noisy speech signals emulated to be uttered in four types of noisy environments: bus (BUS), cafe (CAF), pedestrian area (PED), and street junction (STR). We randomly chose 25 utterances for each environment (100 utterances in total). These simulated utterances were emulated to be recorded with a tablet with 6 microphones by convolving impulse responses obtained from real recordings with clean signals and adding environmental noise. We selected five channels ($M = 5$) excluding the second channel because of its orientation on the back side of the tablet, in contrast to the other five microphones placed on the front side. We used short-time Fourier transform (STFT) with a shifting interval of 256 points and a window length of 1024 points ($F = 513$). The average number of time frames was $T = 379$.

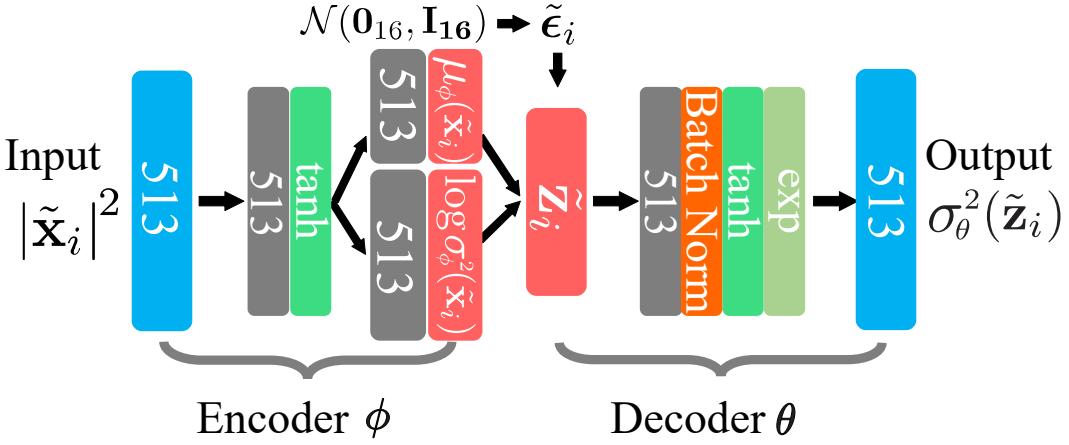


Figure 3.3: The VAE for clean speech spectra.

Performance Measures

The performance of speech enhancement was measured in terms of the signal-to-distortion ratio (SDR) [136, 137]. For comparison with conventional methods, the perceptual evaluation of speech quality (PESQ) [138] and the short-time objective intelligibility (STOI) [139] were also calculated. The fifth channel of the enhanced speech spectra $\{\mathbf{x}_{0ft}^{\mathrm{FR/R1}}\}_{f=1,t=1}^{F,T}$ was compared with the ground-truth clean speech spectra because the fifth microphone was considered to be the closest to the mouth of a speaker.

Pretraining Configurations

The deep speech prior described in Section 3.1.1 was trained in advance from clean speech data in a variational autoencoding manner as described in Section 3.4.3. The VAE had an inference network (encoder) parameterized by $\boldsymbol{\phi}$ and a generation network (decoder) parameterized by $\boldsymbol{\theta}$, as shown in Fig. 3.3. The architecture of the VAE was similar to that proposed in [41]. The dimensions of the observed and latent spaces were $F = 513$ and $D = 16$, respectively. We used the WSJ-0 corpus [140] containing clean speech signals of about 15 hours. The speakers of the WSJ-0 corpus were disjoint with those of the test data. The power spectrogram of each utterance was scaled such that the average power was equal to a random number $\rho \sim \mathrm{Gamma}(2, 2)$, which has the expectation value 1.

Optimization Configurations

The number of iterations was set to 100. For MNMF-DSP, $\mathbf{U}$, $\mathbf{V}$, $\mathbf{W}$, $\mathbf{H}$, and $\mathbf{G}$ were updated simultaneously and $\mathbf{Z}$ was then updated in each iteration. For the ILRMA-DSP, $\mathbf{W}$, $\mathbf{H}$, $\mathbf{U}$, $\mathbf{V}$, $\mathbf{Z}$, and $\mathbf{D}$ were updated in this order. When the sampling method was used for optimizing $\mathbf{Z}$, the variance of the proposal distribution was set to $\xi = 10^{-4}$ and $\mathbf{Z}$ was sampled 50 times per iteration. When the backpropagation method was used, $\mathbf{Z}$ was updated 30 times per iteration by using the Adam optimizer [141] with a learning rate of 0.001.

3.5.2 Evaluation of Model Complexities

We investigated the best model complexities of MNMF-DSP and ILRMA-DSP by changing the number of noise sources N and the number of noise basis spectra K.

Experimental Conditions

For MNMF-DSP, we tested all possible combinations of $K = 2^l$ $(l = 0, \ldots, 10)$ and $N = 1, 2, 3, 4$. For ILRMA-DSP, we changed only K because $N = 4$ must hold under a determined condition with $M = 5$ (one speech source and four noise sources). We used the sampling method for optimizing the latent variables $\mathbf{Z}$ and the observation-based method given by Eq. (3.61) or Eq. (3.65) for initializing the spatial parameters $\mathbf{G}$ or $\mathbf{D}$, respectively.

Experimental Results

Table 3.1-(a) shows the average SDRs over the 100 utterances obtained by MNMF-DSP. The average SDR of the input noisy signals (the fifth channel) was 7.5 dB. Regardless of the choice of N, the performance converged to around 18.7 dB as K increased. This might be because most noise sources in the test dataset were diffusive. If there are multiple directional noise sources, it would be necessary to carefully choose N. Note that when $K \geq T$, the low-rank assumption on the PSDs of noise is considered to have no effect in theory because the noise model is capable of perfectly fitting any PSDs. In reality, the performance was

Table 3.1: The average SDRs [dB] for 100 noisy speech signals in four different environments.

(a) MNMF-DSP

# of noise sources N	Number of noise bases K										
	1	2	4	8	16	32	64	128	256	512	1024
1	16.4	17.1	17.5	17.9	18.1	18.4	18.6	18.7	18.7	18.7	18.7
2	17.3	17.7	18.0	18.2	18.6	18.7	18.8	18.8	18.8	18.8	18.8
3	17.6	18.0	18.3	18.5	18.6	18.7	18.7	18.7	18.8	18.7	18.7
4	17.9	18.1	18.2	18.5	18.6	18.6	18.6	18.6	18.7	18.6	18.6

(b) ILRMA-DSP

# of noise sources N	Number of noise bases K										
	1	2	4	8	16	32	64	128	256	512	1024
4	16.2	16.3	16.2	16.2	16.0	15.8	15.7	15.6	15.5	15.3	15.1

not degraded even when $K = 1024$. This result raised a question whether the low-rank assumption, which is useful in MNMF, is still necessary in the proposed model. To answer this question, the effectiveness of the low-rank assumption was verified in Section 3.5.3. Table 3.2-(a) shows the elapsed times per iteration for processing noisy speech signals of 2 [s] on a workstation with Intel Xeon W-2145 (3.70 GHz). Considering both the performance and the computational cost, the combination of $N = 1$ and $K = 64$ can be regarded as best.

Table 3.1-(b) shows the average SDRs obtained by ILRMA-DSP. The performance was maximized when $K = 2$ and it monotonically decreased as K increased. Because the rank-1 spatial model is incapable of precisely representing realistic sound propagation processes, the source models (speech and noise models) play an influential role for speech enhancement. In each iteration, the noise model fits the current estimate of the noise spectra $\{|\tilde{s}_{nft}|^2\}_{f,t=1}^{F,T}$ given by Eq. (3.42) using the demixing matrices $\mathbf{D}$. The noise model based on NMF with large K overfit the imperfect estimate of the noise spectra in a few iterations before $\mathbf{D}$ was fully optimized. When the noise model with $K = 256$ was updated once per four iterations, the average SDR was improved to 16.1 dB.

Table 3.2: The elapsed times [s] per iteration for processing multichannel noisy speech signals of 2 [s].

(a) MNMF-DSP

# of noise sources N	Number of noise bases K										
	1	2	4	8	16	32	64	128	256	512	1024
1	0.97	0.98	0.99	0.97	0.98	0.99	1.00	1.09	1.22	1.49	2.02
2	1.15	1.15	1.13	1.11	1.15	1.14	1.30	1.41	1.70	2.18	3.27
3	1.34	1.34	1.34	1.35	1.36	1.43	1.53	1.72	2.08	2.87	4.45
4	1.50	1.51	1.50	1.49	1.51	1.63	1.73	2.00	2.51	3.57	5.74

(b) ILRMA-DSP

# of noise sources N	Number of noise bases K										
	1	2	4	8	16	32	64	128	256	512	1024
4	0.28	0.28	0.28	0.29	0.30	0.33	0.40	0.54	0.85	1.44	2.73

3.5.3 Evaluation of Low-Rank Modeling

We investigated the effectiveness of the low-rank assumption on the noise PSDs. The sampling method was used for optimizing the latent variables $\mathbf{Z}$.

Experimental Conditions

We tested three variants of the noise model in MNMF-DSP with $N = 1$.

1. High-rank model: $K = T$. $\mathbf{W}$ and $\mathbf{H}$ were initialized by using Eq. (3.57) and Eq. (3.58) and then iteratively updated by using Eq. (3.29) and Eq. (3.30).
2. 1-on-1 model: This model was the same as the high-rank model except that $\mathbf{H}$ was initialized as follows:

$$\begin{cases} h_{1kt} \sim \mathrm{Gamma}\left(\alpha_0, \frac{\alpha_0}{\mathbb{E}_{\mathrm{emp}}[|x|^2]} \frac{1}{FM}\right) & (k = t), \\ h_{1kt} = 0 & (k \neq t). \end{cases} \tag{3.79}$$

   Since $h_{1kt} = 0$ $(k \neq t)$ was kept in Eq. (3.30), the K bases correspond to the T frames one by one.
3. Non-factorized model: The NMF-based noise model was removed from the proposed model, i.e., the noise PSDs $\{\lambda_{1ft}\}_{f=1,t=1}^{F,T}$ in Eq. (3.8) were directly

Table 3.3: The average SDRs [dB] and log-likelihoods obtained by the three variants of MNMF-DSP.

Noise model	High-rank	1-on-1	Non-factorized
SDR [dB]	18.7	15.8	16.2
Log-likelihood	1.64×10^6	1.67×10^6	1.67×10^6

estimated. An updating rule can be obtained as follows:

$$\lambda_{1ft} \leftarrow \lambda_{1ft} \sqrt{\frac{\mathrm{tr}\left(\mathbf{G}_{1f}\mathbf{Y}_{ft}^{-1}\mathbf{X}_{ft}\mathbf{Y}_{ft}^{-1}\right)}{\mathrm{tr}\left(\mathbf{G}_{1f}\mathbf{Y}_{ft}^{-1}\right)}}. \tag{3.80}$$

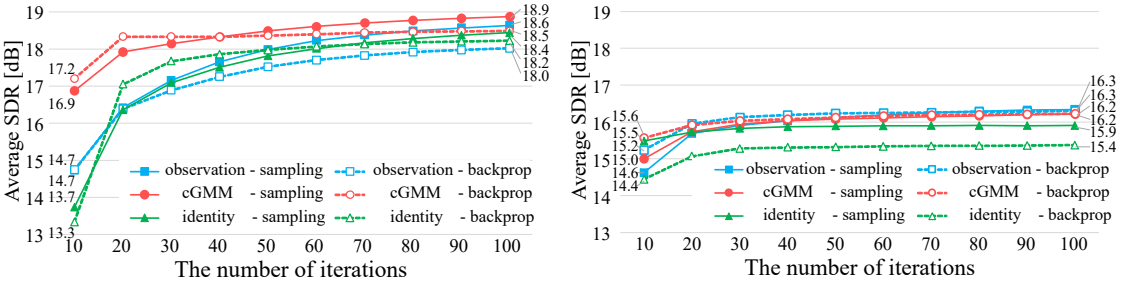
λ_{1ft} was initialized as $\lambda_{1ft} = w_{1tf}h_{1tt}$, where $\mathbf{W}$ and $\mathbf{H}$ were initialized as in the 1-on-1 model.

Experimental Results

Table 3.3 shows the average SDRs and log-likelihoods obtained by the three models. While the 1-on-1 model and the non-factorized model were better than the high-rank model in terms of the log-likelihood, the high-rank model attained the best SDR. Since the architecture of the high-rank model was the same as that of the 1-on-1 model, the high-rank model was considered to get stuck in local optima in which the noise PSDs were approximated as low-rank matrices consisting of a fewer number of bases. This indicates that when $K \geq T$ in Table 3.1-(a), the low-rank constraint on the noise PSDs was still effective. Comparing the SDR (16.2 dB) obtained by the non-factorized model with that (18.6 dB) obtained by the best MNMF-DSP with $N = 1$ and $K = 64$, the low-rank modeling can be said to be effective.

3.5.4 Evaluation of Optimization and Initialization Methods

We investigated the initialization sensitivity and optimization difficulty of MNMF-DSP and ILRMA-DSP.



(a) MNMF-DSP (b) ILRMA-DSP

Figure 3.4: The evolutions of average SDRs [dB] over iterations. The dotted lines indicate the SDRs obtained by the backpropagation method and the solid lines indicate the SDRs obtained by the sampling method.

Experimental Conditions

Considering Tables 3.1 and 3.2, we used MNMF-DSP with $N = 1$ and $K = 64$ and ILRMA-DSP with $K = 2$ as the best performing models with the reasonable computational costs. To optimize $\mathbf{Z}$ in MNMF-DSP, the sampling or backpropagation method was used (Section 3.2.2). To initialize $\mathbf{G}$, the identity-, observation-, or cGMM-based method given by Eqs. (3.60), (3.61), or (3.62), respectively, was used (Section 3.4.1). To optimize $\mathbf{Z}$ in ILRMA-DSP, on the other hand, the sampling or backpropagation method was used (Section 3.3.2). To initialize $\mathbf{D}$, the identity-, observation-, or cGMM-based method given by Eqs. (3.64), (3.65), or (3.66) was used (Section 3.4.2). In total, we tested six configurations for each model.

Experimental Results

Fig. 3.4-(a) shows the SDR evolutions over iterations obtained by the six configurations of MNMF-DSP. The combination of the sampling-based optimization and the cGMM-based initialization attained the best SDR of 18.9 dB. Regardless of an initialization method, the sampling method was slightly better than the backpropagation method in terms of the performance obtained after sufficiently many iterations. The backpropagation method converged faster to the affordable performance than the sampling method. When the same optimization method was used, the performance difference between the initialization methods was smaller than 0.5 dB. This indicates that our MNMF-DSP is insensitive to the

initialization of $\mathbf{G}$ because the deep speech prior plays an influential role even before $\mathbf{G}$ is not fully optimized. In fact, our model can work even in a single-channel scenario without spatial information [40]. This is a noticeable advantage of the proposed method over MNMF that heavily relies on $\mathbf{G}$ for speech enhancement.

Fig. 3.4-(b) shows the SDR evolutions over iterations obtained by the six configurations of ILRMA-DSP. The use of the observation- or cGMM- based initialization method reached the SDR of 16.3 dB or 16.2 dB, respectively, regardless of the optimization strategy. The backpropagation method tended to converge faster than the sampling method. When the identity-based initialization method was used, the backpropagation method underperformed the sampling method. The initial values of $\mathbf{Z}$ given by the encoder ϕ of the VAE were considered to be close to optimal values. In the backpropagation method, however, $\mathbf{Z}$ was quickly adapted to the inaccurate estimate of the speech spectra $\mathbf{s}_{ft}$ given by Eq. (3.42) before the demixing matrices $\mathbf{D}$ were fully optimized.

3.5.5 Key Findings

Considering the experimental results shown in Section 3.5.2 and Section 3.5.4, we summarize recommended configurations. In general, it is recommended to use the full-rank model with $N = 1$, $K = 64$, the observation-based initialization method given by Eq. (3.61), and the sampling-based optimization method. To squeeze the performance and accelerate the convergence in exchange of the additional implementation cost, one can use the cGMM-based initialization method given by Eq. (3.62) instead of the observation-based initialization method. If the computational cost is a main concern, one may use ILRMA-DSP with $K = 2$, the observation-based initialization method given by Eq. (3.65), and the backpropagation-based optimization method.

3.5.6 Comparison with State-of-the-Art Methods

We compared the proposed semi-blind method with the state-of-the-art blind, semi-blind, and supervised methods in terms of the SDR, PESQ, and STOI.

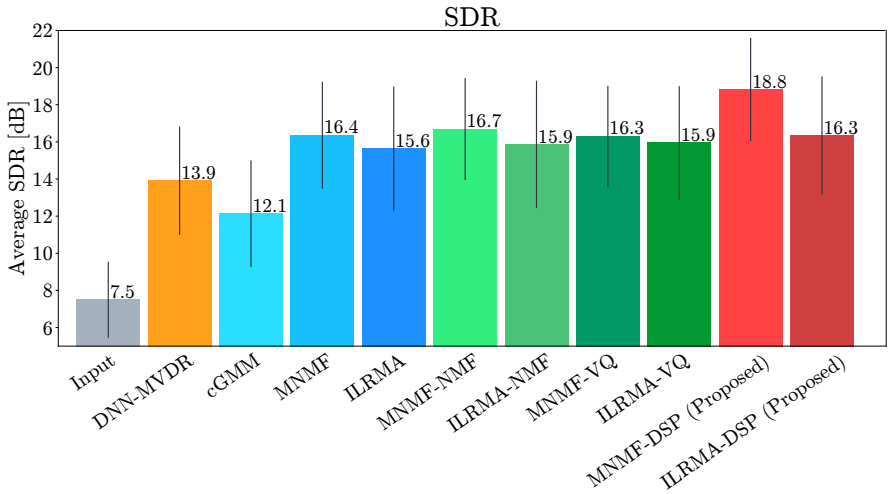


Figure 3.5: The average SDRs obtained by the 11 methods.

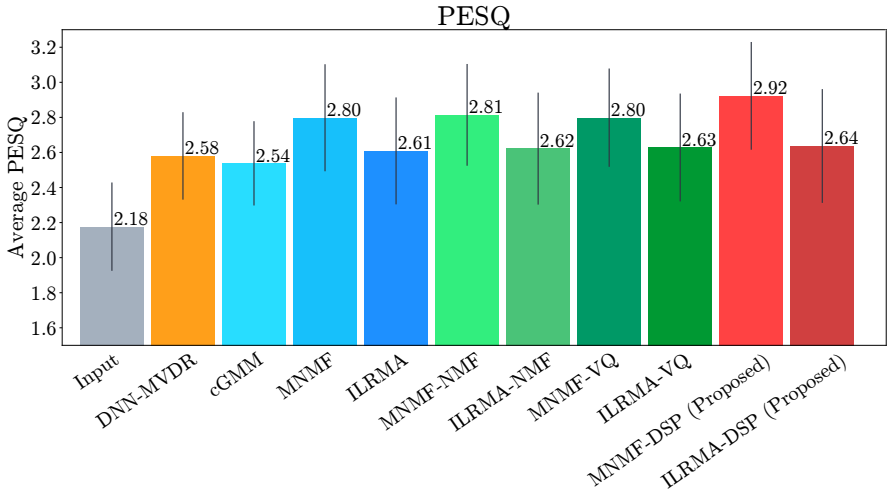


Figure 3.6: The average PESQs obtained by the 11 methods.

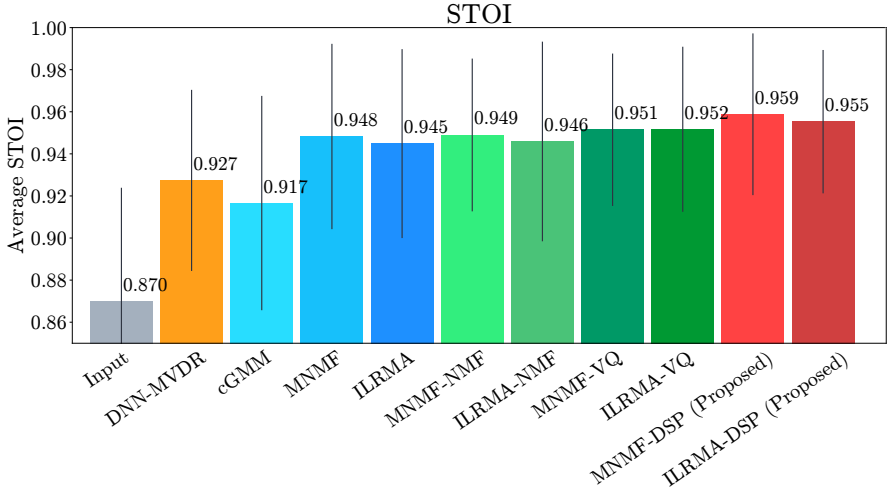


Figure 3.7: The average STOIs obtained by the 11 methods.

Experimental Conditions

We used MNMF-DSP with $N = 1$ and $K = 64$ initialized by the cGMM-based method given by Eq. (3.62) and ILRMA-DSP with $N = 4$ and $K = 2$ initialized by the observation-based method given by Eq. (3.65). In both models, the sampling method was used for estimating $\mathbf{Z}$.

- Unsupervised methods: We tested MNMF [27], ILRMA [28], and cGMM [89]. MNMF and ILRMA had the same architectures as the proposed MNMF-DSP and ILRMA-DSP, respectively, except that an NMF-based low-rank model was used for speech instead of the DNN-based model. The number of noise sources N, that of speech bases K_s, that of noise bases K_n, and the initialization strategy were experimentally optimized. We used MNMF with $N = 1$, $K_s = 8$, and $K_n = 256$ initialized by the cGMM-based method and ILRMA with $N = 4$, $K_s = 8$, and $K_n = 1$ initialized by the observation-based method. We also tested a weighted delay-and-sum (DS) beamforming called *beamformit* [142] and found that the average SDR of the enhanced speech was 6.3 dB.
- Semi-blind methods: For fair comparison with the proposed semi-blind method, we also tested semi-blind versions of MNMF and ILRMA. K_s speech bases were estimated in advance by using NMF [8] or vector quantization (VQ) [143] based on the Itakura-Saito (IS) divergence (called IS-NMF and IS-VQ, respectively) for the clean speech data of the WSJ-0 corpus [140]. IS-VQ iterated two steps; 1) given codebooks (bases), each speech spectrum in the training dataset were clustered into the nearest codebook based on the IS divergence, and 2) each codebook was updated to the average of the spectra assigned to the codebook. In the speech enhancement phase, while the speech bases were fixed, the other parameters were updated as in the proposed method. We conducted a preliminary comparative experiment using $K_s = 2^l$ $(l = 0, \ldots, 8)$ and decided to use MNMF based on IS-NMF (MNMF-NMF) with $N = 1$, $K_s = 4$, and $K_n = 256$ and MNMF based on IS-VQ (MNMF-VQ) with $N = 1$, $K_s = 8$, and $K_n = 256$ initialized by the

cGMM-based method, and ILRMA based on IS-NMF (ILRMA-NMF) with $N = 4$, $K_s = 16$, and $K_n = 1$ and ILRMA based on IS-VQ (ILRMA-VQ) with $N = 4$, $K_s = 256$, and $K_n = 2$ initialized by the observation-based method.

- Supervised method: We tested a DNN-based beamforming method. To estimate speech masks $\{\omega_{ft}\}_{f=1,t=1}^{F,T}$, a feed-forward DNN was trained by using the training dataset of CHiME3 that contains pairs of multichannel noisy speech signals and ground-truth clean speech signals. We extracted three kinds of acoustic features as the input to the DNN. At each time t, the log of the outputs of 100-channel mel-scale filter banks (LMFBs) was computed from the magnitude spectrogram of the fifth channel and LMFBs were stacked over 11 frames from time $t-5$ to $t+5$. The $(M-1)$-dimensional inter-channel level and phase differences (ILDs and IPDs) were also calculated at each time t as proposed in [144]. The DNN was trained such that the cross-entropy loss between ideal binary masks and estimated masks was minimized. To use the minimum variance distortionless response (MVDR) beamforming [21], the steering vector of speech $\mathbf{a}_{0f}$ and the SCM of noise $\mathbf{G}_{1f}$ at frequency f are given by

$$\begin{cases} \mathbf{a}_{0f} = \mathcal{PE}\left(\sum_{t=1}^{T} \omega_{ft}\mathbf{X}_{ft}\right), \\ \mathbf{G}_{1f} = \sum_{t=1}^{T}(1-\omega_{ft})\mathbf{X}_{ft}, \end{cases} \tag{3.81}$$

where $\mathcal{PE}(\cdot)$ indicates a normalized eigenvector that corresponds to the first principal component of a matrix. The demixing filter $\mathbf{d}_{0f}$ at frequency f is given by

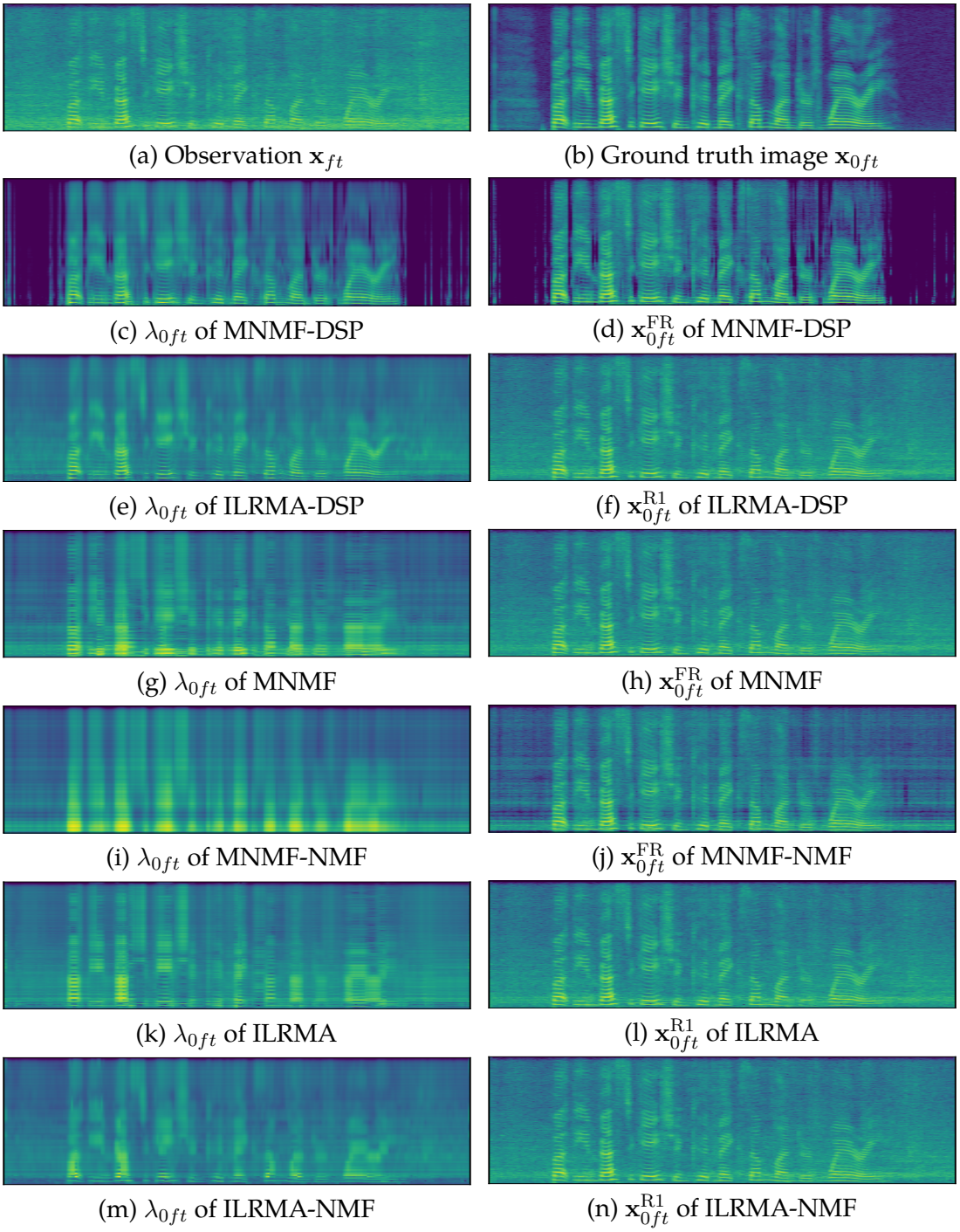
$$\mathbf{d}_{0f} = \frac{\mathbf{G}_{1f}^{-1}\mathbf{a}_{0f}}{\mathbf{a}_{0f}^{\mathsf{H}}\mathbf{G}_{1f}^{-1}\mathbf{a}_{0f}}. \tag{3.82}$$

The speech image was estimated as follows:

$$\mathbf{x}_{0ft}^{\mathrm{MVDR}} = \mathbf{a}_{0f}\tilde{s}_{0ft} = \mathbf{a}_{0f}\mathbf{d}_{0f}^{\mathsf{H}}\mathbf{x}_{ft}. \tag{3.83}$$

Experimental Results

Figs. 3.5, 3.6, and 3.7 show the average SDRs, PESQs, and STOIs, respectively. MNMF-DSP performed best in all measures, and Welch's t-test with a 0.05



(a) Observation $\mathbf{x}_{ft}$

(b) Ground truth image $\mathbf{x}_{0ft}$

(c) λ_{0ft} of MNMF-DSP

(d) $\mathbf{x}_{0ft}^{\mathrm{FR}}$ of MNMF-DSP

(e) λ_{0ft} of ILRMA-DSP

(f) $\mathbf{x}_{0ft}^{\mathrm{R1}}$ of ILRMA-DSP

(g) λ_{0ft} of MNMF

(h) $\mathbf{x}_{0ft}^{\mathrm{FR}}$ of MNMF

(i) λ_{0ft} of MNMF-NMF

(j) $\mathbf{x}_{0ft}^{\mathrm{FR}}$ of MNMF-NMF

(k) λ_{0ft} of ILRMA

(l) $\mathbf{x}_{0ft}^{\mathrm{R1}}$ of ILRMA

(m) λ_{0ft} of ILRMA-NMF

(n) $\mathbf{x}_{0ft}^{\mathrm{R1}}$ of ILRMA-NMF

Figure 3.8: Comparison of speech enhancement methods. The observation, ground truth, and separated speech show the 5th channel only.

significance level showed the significant difference of SDR and PESQ between MNMF-DSP and the other methods, although the differences of STOI between MNMF-DSP and MNMF, MNMF-NMF, MNMF-VQ, ILRMA-VQ, or ILRMA-DSP were not significant. Although, MNMF is generally known to often underperform ILRMA because of the strong initialization sensitivity [28], in this experiment MNMF (16.4 dB) outperformed ILRMA (15.6 dB) because the cGMM-based method given by Eq. (3.62) provided a good initial estimate of $\mathbf{G}$. When the observation-based method given by Eq. (3.61) was used for MNMF as in ILRMA, the SDR was drastically degraded (12.6 dB). Fig. 3.8 shows examples of the noisy spectra $\{\mathbf{x}_{ft}\}_{f=1,t=1}^{F,T}$ in the BUS environment, the ground-truth speech image $\{\mathbf{x}_{0ft}\}_{f=1,t=1}^{F,T}$, the estimated speech PSDs $\{\lambda_{0ft}\}_{f=1,t=1}^{F,T}$ and the separated speech spectra $\{\mathbf{x}_{0ft}^{\mathrm{FR/R1}}\}_{f=1,t=1}^{F,T}$ obtained by MNMF-DSP (19.2 dB), ILRMA-DSP (14.9 dB), MNMF (15.3 dB), ILRMA (13.4 dB), MNMF-NMF (16.1 dB), ILRMA-NMF (14.7 dB), MNMF-VQ (15.8 dB), and ILRMA-VQ (14.6 dB). This clearly showed that the deep speech prior is better at representing the characteristic structures of speech PSDs than the NMF-based low-rank model. In semi-blind MNMF and ILRMA, the numbers of speech bases were determined to maximize the SDRs, but were too low to precisely represent speech PSDs. To confirm the effectiveness of the deep speech prior, we also tested MNMF initialized with $\mathbf{G}$ estimated by MNMF-DSP, and the average SDR after 100 iterations was 17.5 dB. This result indicates that the high representation power of the deep speech prior was useful for not only alleviating the initialization sensitivity but also improving the performance.

We also compared the proposed method with its original single-channel version [40]. The SDR of the optimally-tuned single-channel method was 11.9 dB. This indicates that the proposed MNMF-DSP and ILRMA-DSP successfully utilize the spatial information. Comparing ILRMA-DSP with MNMF-DSP, however, while MNMF-DSP successfully suppressed the noise components, the enhanced speech spectra obtained by ILRMA-DSP as well as those obtained by ILRMA were still noisy. This indicates that the idealized rank-1 spatial model based on the time-invariant demixing matrices has a performance limitation in speech

enhancement.

3.6 Summary

This chapter presented a semi-blind multichannel speech enhancement method that integrates a DNN-based generative model of speech spectra, an NMF-based generative model of noise spectra, and a full-rank or rank-1 spatial model in a unified probabilistic model. The full-rank and rank-1 versions of the proposed method, called MNMF-DSP and ILRMA-DSP, are extensions of MNMF [27] and ILRMA [28], respectively, i.e., an NMF-based model for one of sources is replaced with the deep speech prior capable of precisely representing the PSDs of clean speech. An advantage of our method is that only clean speech data are used for training the deep speech prior. The speech prior can generalize well to unseen speech spectra and the low-rank noise model and the spatial model can adapt to unseen acoustic environments. We showed that MNMF-DSP significantly outperformed the rank-1 counterpart, the blind and semi-blind versions of MNMF and ILRMA, and the supervised DNN-based beamforming method in terms of the SDR, PESQ, and STOI, because of the high representation power of the deep speech prior. We also showed that MNMF-DSP is less sensitive to initialization and is less likely to get stuck in local optima than MNMF. Recently, this approach has been extended for speech separation by using the deep speech generative models for all sources [45, 91, 94, 145].

Although MNMF-DSP achieved good performance, its main limitation is high computational cost due to the repeated heavy operations such as inversion of the SCMs. In chapter 4, we introduce a jointly-diagonalizable full-rank spatial model to reduce computational cost without degrading its performance.

Chapter 4

Fast Multichannel Speech Separation Based on a Jointly-Diagonalizable Spatial Model

4.1 Introduction

MNMF based on the full-rank spatial model is known to be computationally expensive and sensitive to the initialization of the parameters, and its performance is often lower than that of ILRMA based on the rank-1 spatial model. In Chapter 3, the extensions of MNMF and ILRMA using the DNN-based speech model called MNMF-DSP and ILRMA-DSP were proposed. MNMF-DSP is less sensitive to the initialization due to the powerful DNN-based speech model, and it outperforms ILRMA and ILRMA-DSP. In this chapter, we tackle the problem about the high computational cost of MNMF.

As an intermediate BSS method between MNMF and ILRMA, first, we propose a computationally-efficient variant of MNMF called FastMNMF1 (Fig. 4.1) that restricts all source SCMs of each frequency bin to jointly-diagonalizable (JD) yet full-rank matrices [83]. Note that another FastMNMF1 based on the same formulation had been developed independently and concurrently [44]. To estimate the SCMs, we use a convergence-guaranteed iterative projection (IP) algorithm [70], while [44] uses a fixed-point iteration (FPI) algorithm without

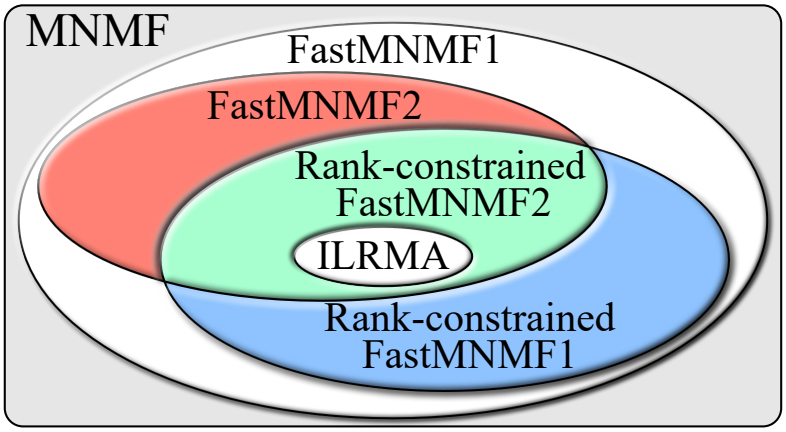


Figure 4.1: The relations between MNMF, ILRMA, and the proposed FastMNMF, including its variants.

convergence guarantee. Although FastMNMF1 is almost as fast as ILRMA, its initialization sensitivity inherited from MNMF is still an open problem. For speech enhancement, DNN-based speech model can be used to alleviate the sensitivity.

To reduce the initialization sensitivity of FastMNMF1 for source separation, we propose a well-behaved constrained version of FastMNMF1 called FastMNMF2 that shares the directional feature of each source over all frequency bins. The JD spatial model of FastMNMF1 assumes that in each frequency bin, the SCM of each source is represented by the weighted sum of M common rank-1 SCMs, which are expected to correspond to M directions. While the weights of each source vary over frequency bins in FastMNMF1, they are shared over all frequency bins in FastMNMF2. This directivity-aware spatial model would mitigate the permutation problem, which has mainly been tackled by improving the source model.

To explicitly consider the directivity or diffuseness of each source, we further propose a rank-constrained version of FastMNMF1 or FastMNMF2 (collectively called FastMNMF) that enables us to individually specify the ranks of SCMs. When one or more people are talking in a noisy environment, for example, our goal is to separate an observed mixture into directional speech sources and diffuse noise sources. Such speech enhancement or separation can be achieved

by initializing the weights of speech and noise sources to one-hot and all-one vectors, respectively, because the number of non-zero weights indicates the rank of an SCM. Through the iterative optimization, the speech SCMs are kept to rank-1 matrices and the noise SCMs to full-rank matrices thanks to the nature of the multiplicative update algorithm. If the SCMs of all M sources are restricted to rank-1 matrices in a determined case, rank-constrained FastMNMF reduces to ILRMA.

4.2 MNMF with a Jointly-Diagonalizable Spatial Model (FastMNMF1)

We formulate the probabilistic model of FastMNMF1 using the low-rank source model and the jointly-diagonalizable (JD) full-rank spatial model and then derive an efficient parameter estimation algorithm based on iterations of nonnegative tensor factorization (NTF) and IP.

4.2.1 Formulation

In order to reduce the degree of freedom, we assume that the SCMs of N sources $\{\mathbf{G}_{nf}\}_{n=1}^{N}$ are JD as follows:

$$\forall n \ \mathbf{G}_{nf} = \mathbf{Q}_f^{-1}\mathrm{Diag}(\tilde{\mathbf{g}}_{nf})\mathbf{Q}_f^{-\mathsf{H}}, \tag{4.1}$$

where $\tilde{\mathbf{g}}_{nf} \triangleq [\tilde{g}_{nf1}, \ldots, \tilde{g}_{nfM}] \in \mathbb{R}_+^M$ is a nonnegative vector, and $\mathbf{Q}_f \triangleq [\mathbf{q}_{f1}, \ldots, \mathbf{q}_{fM}]^{\mathsf{H}} \in \mathbb{C}^{M\times M}$ is a non-singular matrix called a *diagonalizer*, which is not limited to a unitary matrix. Substituting Eq. (4.1) into Eq. (1.6), we have

$$\mathbf{Q}_f\mathbf{x}_{ft} \sim \mathcal{N}_{\mathbb{C}}\left(\mathbf{0}, \sum_{n=1}^{N}\lambda_{ftn}\mathrm{Diag}(\tilde{\mathbf{g}}_{nf})\right). \tag{4.2}$$

This means that the elements of $\mathbf{Q}_f\mathbf{x}_{ft}$ are all independent. Regarding $\mathbf{Q}_f\mathbf{x}_{ft}$ as observed data, MNMF for $\mathbf{Q}_f\mathbf{x}_{ft}$ reduces to nonnegative tensor factorization (NTF) for the PSDs of $\mathbf{Q}_f\mathbf{x}_{ft}$, which can be performed efficiently (Fig. 4.2). The log-likelihood function of the parameters $\mathbf{W}$, $\mathbf{H}$, $\tilde{\mathbf{G}} \triangleq \{\tilde{\mathbf{g}}_{nf}\}_{n,f=1}^{N,F}$, and $\mathbf{Q} \triangleq \{\mathbf{Q}_f\}_{f=1}^{F}$

is then given by

$$\begin{aligned}\log p(\mathbf{X}|\mathbf{W},\mathbf{H},\tilde{\mathbf{G}},\mathbf{Q}) &= \sum_{f,t=1}^{F,T}\log\mathcal{N}_{\mathbb{C}}\left(\mathbf{x}_{ft}\,\middle|\,\mathbf{0},\sum_{n=1}^{N}\lambda_{ftn}\mathbf{Q}_f^{-1}\mathrm{Diag}(\tilde{\mathbf{g}}_{nf})\mathbf{Q}_f^{-\mathsf{H}}\right)\\ &= -\sum_{f,t,m=1}^{F,T,M}\left(\frac{\tilde{x}_{ftm}}{\tilde{y}_{ftm}}+\log\tilde{y}_{ftm}\right)+T\sum_{f=1}^{F}\log\left|\mathbf{Q}_f\mathbf{Q}_f^{\mathsf{H}}\right|+\mathrm{const},\end{aligned} \tag{4.3}$$

where $\tilde{x}_{ftm}\triangleq|\mathbf{q}_{fm}^{\mathsf{H}}\mathbf{x}_{ft}|^2$ and $\tilde{y}_{ftm}\triangleq\sum_{n,k}w_{nkf}h_{nkt}\tilde{g}_{nfm}$.

To avoid the scale ambiguity, we put normalization constraints on $\mathbf{Q}$, $\tilde{\mathbf{G}}$, and $\mathbf{W}$ as follows:

$$\mathbf{q}_{fm}^{\mathsf{H}}\mathbf{q}_{fm}=1, \tag{4.4}$$

$$\sum_{m=1}^{M}\tilde{g}_{nfm}=1, \tag{4.5}$$

$$\sum_{f=1}^{F}w_{nkf}=1. \tag{4.6}$$

4.2.2 Optimization

Our goal is to jointly estimate $\mathbf{W}$, $\mathbf{H}$, $\tilde{\mathbf{G}}$, and $\mathbf{Q}$ such that the log-likelihood function given by Eq. (4.3) is maximized. Because Eq. (4.3) has the same form as the log-likelihood function of ILRMA given by Eq. (3.46), we can derive a convergence-guaranteed optimization algorithm based on iterations of NTF and IP in the same way as ILRMA, which is based on iterations of NMF and IP.

Because the first term of Eq. (4.3) is the negative Itakura-Saito (IS) divergence between $\tilde{x}_{ftm}$ and $\tilde{y}_{ftm}$, the maximization of the log-likelihood with respect to $\mathbf{W}$, $\mathbf{H}$, and $\tilde{\mathbf{G}}$ is equivalent to NTF. The multiplicative update (MU) rules for $\mathbf{W}$, $\mathbf{H}$, and $\tilde{\mathbf{G}}$ are given by

$$w_{nkf}\leftarrow w_{nkf}\sqrt{\frac{\sum_{t,m=1}^{T,M}h_{nkt}\tilde{g}_{nfm}\tilde{x}_{ftm}\tilde{y}_{ftm}^{-2}}{\sum_{t,m=1}^{T,M}h_{nkt}\tilde{g}_{nfm}\tilde{y}_{ftm}^{-1}}}, \tag{4.7}$$

$$h_{nkt}\leftarrow h_{nkt}\sqrt{\frac{\sum_{f,m=1}^{F,M}w_{nkf}\tilde{g}_{nfm}\tilde{x}_{ftm}\tilde{y}_{ftm}^{-2}}{\sum_{f,m=1}^{F,M}w_{nkf}\tilde{g}_{nfm}\tilde{y}_{ftm}^{-1}}}, \tag{4.8}$$

$$\tilde{g}_{nfm} \leftarrow \tilde{g}_{nfm} \sqrt{\frac{\sum_{t,k=1}^{T,K} w_{nkf} h_{nkt} \tilde{x}_{ftm} \tilde{y}_{ftm}^{-2}}{\sum_{t,k=1}^{T,K} w_{nkf} h_{nkt} \tilde{y}_{ftm}^{-1}}}. \tag{4.9}$$

As in IVA [70] and ILRMA [28], the IP rules of $\mathbf{Q}_f$ are given by

$$\mathbf{V}_{fm} \triangleq \frac{1}{T} \sum_{t=1}^{T} \mathbf{X}_{ft} \tilde{y}_{ftm}^{-1}, \tag{4.10}$$

$$\mathbf{q}_{fm} \leftarrow (\mathbf{Q}_f \mathbf{V}_{fm})^{-1} \mathbf{e}_m, \tag{4.11}$$

$$\mathbf{q}_{fm} \leftarrow (\mathbf{q}_{fm}^{\mathsf{H}} \mathbf{V}_{fm} \mathbf{q}_{fm})^{-\frac{1}{2}} \mathbf{q}_{fm}, \tag{4.12}$$

where $\mathbf{e}_m$ is a one-hot vector whose m-th element is 1. The diagonalizer $\mathbf{Q}_f$ is estimated such that the M *components* of $\{\mathbf{Q}_f \mathbf{x}_{ft}\}_{f,t=1}^{F,T}$ become independent. In ILRMA under a determined condition ($N = M$), a demixing matrix $\mathbf{D}_f$ is estimated such that the M *sources* of $\{\mathbf{D}_f \mathbf{x}_{ft}\}_{f,t=1}^{F,T}$ become independent. Therefore, $\mathbf{Q}_f$ and $\mathbf{D}_f$ are estimated in almost the same way, and expected to play a similar role.

To satisfy the normalization constraints given by Eqs. (4.4), (4.5), and (4.6), we adjust the scales of $\mathbf{Q}$, $\tilde{\mathbf{G}}$, and $\mathbf{W}$ in this order in each iteration as follows:

$$\mu_{fm} \triangleq \mathbf{q}_{fm}^{\mathsf{H}} \mathbf{q}_{fm}, \quad \begin{cases} \mathbf{q}_{fm} \leftarrow \mu_{fm}^{-\frac{1}{2}} \mathbf{q}_{fm}, \\ \tilde{g}_{nfm} \leftarrow \mu_{fm}^{-1} \tilde{g}_{nfm}, \end{cases} \tag{4.13}$$

$$\phi_{nf} \triangleq \sum_{m=1}^{M} \tilde{g}_{nfm}, \quad \begin{cases} \tilde{g}_{nfm} \leftarrow \phi_{nf}^{-1} \tilde{g}_{nfm}, \\ w_{nkf} \leftarrow \phi_{nf} w_{nkf}, \end{cases} \tag{4.14}$$

$$\nu_{nk} \triangleq \sum_{f=1}^{F} w_{nkf}, \quad \begin{cases} w_{nkf} \leftarrow \nu_{nk}^{-1} w_{nkf}, \\ h_{nkt} \leftarrow \nu_{nk} h_{nkt}. \end{cases} \tag{4.15}$$

4.2.3 Separation

To estimate the source images $\mathbf{x}_{ftn} \in \mathbb{C}^M$, we use a Wiener filtering given by $\mathbb{E}[\mathbf{x}_{ftn}|\mathbf{x}_{ft}] = \mathbf{Y}_{ftn} \mathbf{Y}_{ft}^{-1} \mathbf{x}_{ft}$, which can be rewritten using Eq. (4.1) as follows:

$$\begin{aligned} \mathbb{E}[\mathbf{x}_{ftn}|\mathbf{x}_{ft}] &= \mathbf{Y}_{ftn} \mathbf{Y}_{ft}^{-1} \mathbf{x}_{ft} \\ &= \mathbf{Q}_f^{-1} \mathrm{Diag}\left(\frac{\lambda_{ftn} \tilde{\mathbf{g}}_{nf}}{\sum_n \lambda_{ftn} \tilde{\mathbf{g}}_{nf}} \right) \mathbf{Q}_f \mathbf{x}_{ft}. \end{aligned} \tag{4.16}$$

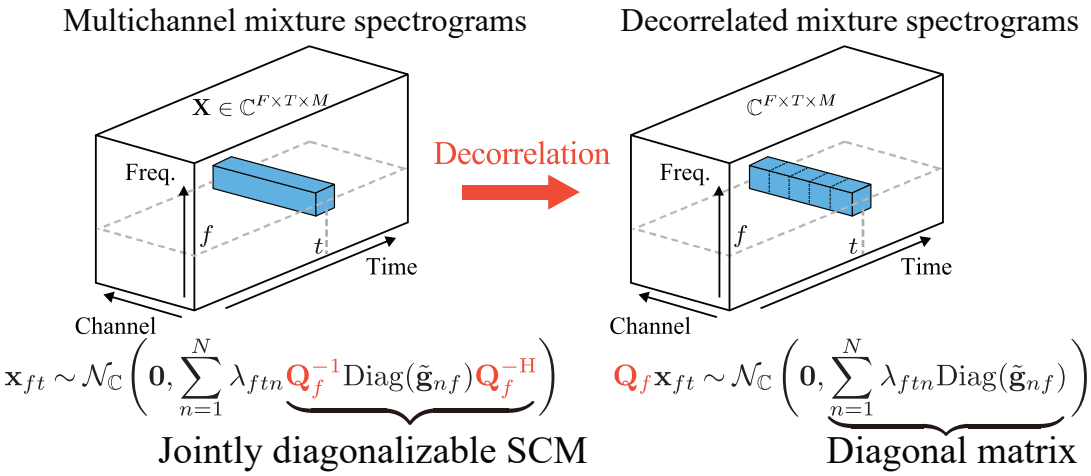


Figure 4.2: The jointly-diagonalizable full-rank spatial model.

4.2.4 Interpretation of Jointly-Diagonalizable Spatial Model

This section discusses the roles of the diagonalizer $\mathbf{Q}$ and the diagonal elements $\tilde{\mathbf{G}}$, which were originally introduced for reducing the computational cost. The joint-diagonalization constraint given by Eq. (4.1) can be rewritten as

$$\mathbf{G}_{nf} = \mathbf{U}_f \mathrm{Diag}(\tilde{\mathbf{g}}_{nf}) \mathbf{U}_f^{\mathsf{H}} = \sum_{m=1}^{M} \tilde{g}_{nfm} \mathbf{u}_{fm} \mathbf{u}_{fm}^{\mathsf{H}}, \tag{4.17}$$

where $\mathbf{U}_f \triangleq [\mathbf{u}_{f1}, \cdots, \mathbf{u}_{fM}] \in \mathbb{C}^{M\times M}$ is a non-singular matrix given by $\mathbf{U}_f = \mathbf{Q}_f^{-1}$. Eq. (4.17) means that $\mathbf{G}_{nf}$ is given as the weighted sum of M rank-1 matrices $\{\mathbf{U}_{fm} \triangleq \mathbf{u}_{fm}\mathbf{u}_{fm}^{\mathsf{H}}\}_{m=1}^{M}$, where the weights are given by $\tilde{\mathbf{g}}_{nf} \triangleq \{\tilde{g}_{nfm}\}_{m=1}^{M}$ (Fig. 4.3). The number of non-zero elements of $\tilde{\mathbf{g}}_{nf}$ represents the rank of $\mathbf{G}_{nf}$.

We clarify how the JD full-rank model relates to the rank-1 spatial model under a determined condition ($N = M$). If $\tilde{\mathbf{g}}_{nf} = \mathbf{e}_n$, $\mathbf{G}_{nf}$ is a rank-1 matrix, and the log-likelihood function of FastMNMF1 given by Eq. (4.3) reduces to that of ILRMA given by Eq. (3.46), where $\mathbf{Q}_f = \mathbf{D}_f$ ($\mathbf{U}_f = \mathbf{A}_f$) and $\tilde{y}_{ftm} = \sum_n \lambda_{ftn}\tilde{g}_{nfm} = \lambda_{ftm}$. This means the JD full-rank spatial model includes the rank-1 spatial model as its special case. Therefore, each $\mathbf{u}_{fm}$ is related to the steering vector of a certain direction, and this is experimentally confirmed in Section 4.7.1. If $\tilde{\mathbf{g}}_{nf} \neq \mathbf{e}_n$, $\tilde{\mathbf{g}}_{nf}$ is considered to represent the weights of M directions for source n. Unlike the rank-1 spatial model, the JD full-rank spatial model is applicable to an

underdetermined condition ($N > M$), but does not work well in practice because at most M directions can be covered.

We clarify the interpretation of the JD full-rank spatial model under an overdetermined condition ($N < M$). Assuming that the reverberation is longer than the window size of short-time Fourier transform (STFT), the image of source n, denoted by $\mathbf{x}_{nft}$, is written by explicitly representing the early reflection with a moving average (MA) model as follows:

$$\mathbf{x}_{nft} = \mathbf{a}_{nf0} s_{nft} + \sum_{l=1}^{L-1} \mathbf{a}_{nfl} s_{n,f,t-l}, \tag{4.18}$$

where L is the length of the impulse response and $\mathbf{a}_{nfl}$ is the STFT coefficients of the impulse response of source n at frequency f and time l. When s_{nft} follows a circularly-symmetric complex Gaussian distribution, $\mathbf{x}_{nft}$ also follows a Gaussian distribution given by

$$\mathbf{x}_{nft} \sim \mathcal{N}_{\mathbb{C}} \left(\mathbf{0}_M, \sum_{l=0}^{L-1} \lambda_{n,f,t-l} \mathbf{a}_{nfl} \mathbf{a}_{nfl}^{\mathsf{H}} \right) \tag{4.19}$$

$$= \mathcal{N}_{\mathbb{C}} \left(\mathbf{0}_M, \lambda_{nft} \sum_{l=0}^{L-1} \frac{\lambda_{n,f,t-l}}{\lambda_{nft}} \mathbf{a}_{nfl} \mathbf{a}_{nfl}^{\mathsf{H}} \right) = \mathcal{N}_{\mathbb{C}} \left(\mathbf{0}_M, \lambda_{nft} \bar{\mathbf{G}}_{nft} \right), \tag{4.20}$$

where $\bar{\mathbf{G}}_{nft} \triangleq \sum_{l=0}^{L-1} \frac{\lambda_{n,f,t-l}}{\lambda_{nft}} \mathbf{a}_{nfl} \mathbf{a}_{nfl}^{\mathsf{H}} \triangleq \sum_{l=0}^{L-1} \alpha_{nftl} \mathbf{a}_{nfl} \mathbf{a}_{nfl}^{\mathsf{H}}$ is a time-varying SCM. Comparing $\mathbf{G}_{nf} = \sum_{m=1}^{M} \tilde{g}_{nfm} \mathbf{u}_{fm} \mathbf{u}_{fm}^{\mathsf{H}}$ and $\bar{\mathbf{G}}_{nft} \approx \sum_{l=0}^{L-1} \alpha_{nftl} \mathbf{a}_{nfl} \mathbf{a}_{nfl}^{\mathsf{H}}$, $\mathbf{u}_{fm}$ is considered to play a similar role as $\mathbf{a}_{nfl}$. Since the vectors $\{\mathbf{u}_{fm}\}_{m=1}^{M}$ are shared over N sources in the JD spatial model, they typically consist of the steering vectors $\{\mathbf{a}_{nf0}\}_{n=1}^{N}$ of the direct paths from source directions and some impulse responses $\mathbf{a}_{n,f,l>0}$ of the predominant reflection paths.

Each source n (*e.g.*, target, noise, or reverberation) is *softly* associated with the vectors $\{\mathbf{u}_{fm}\}_{m=1}^{M}$ according to $\tilde{\mathbf{G}} \triangleq \{\tilde{\mathbf{g}}_{nf}\}_{n=1}^{N}$ at each frequency f. When only directional sources exist without any noise and reverberation, the advantage of the JD full-rank spatial model is limited because each source n would be *hardly* associated with one of $\{\mathbf{u}_{fm}\}_{m=1}^{M}$ as in the rank-1 spatial model. The JD full-rank spatial model is advantageous when adverse non-directional (diffuse) noise and reverberations of non-target sources partially come from the same direction as

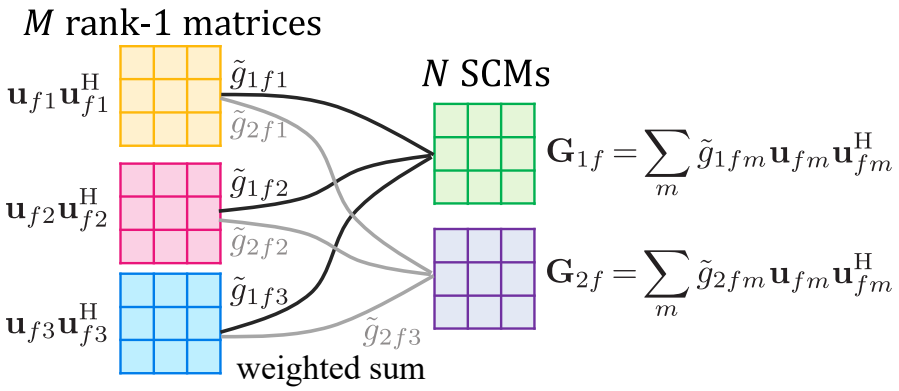


Figure 4.3: Interpretation of the jointly-diagonalizable full-rank spatial model with $M = 3$ and $N = 2$.

target sources. Even under such a condition, FastMNMF can extract a target sound only by suppressing the interfering sounds of the same direction, because $\mathbf{u}_{fm}$ can be shared over multiple sources. In Section 4.7.7, this advantage was shown experimentally.

4.3 MNMF with a Weight-Shared Jointly-Diagonalizable Spatial Model (FastMNMF2)

We formulate the probabilistic model of FastMNMF2 based on the weight-shared version of the JD full-rank spatial model and then derive a modified parameter estimation algorithm.

4.3.1 Formulation

To further reduce the degree of freedom of the JD spatial model, we propose to make the weights $\tilde{\mathbf{g}}_{nf}$ of FastMNMF1 consistent over all frequency bins in light of discussions described in Section 4.2.4. More specifically, Eqs. (4.1) and (4.17) are replaced with

$$\forall n\ \mathbf{G}_{nf} = \mathbf{Q}_f^{-1}\mathrm{Diag}(\tilde{\mathbf{g}}_n)\mathbf{Q}_f^{-\mathsf{H}} = \sum_{m=1}^{M} \tilde{g}_{nm}\mathbf{u}_{fm}\mathbf{u}_{fm}^{\mathsf{H}}, \tag{4.21}$$

where $\tilde{\mathbf{g}}_n \triangleq [\tilde{g}_{n1}, \ldots, \tilde{g}_{nM}] \in \mathbb{R}_+^M$ is a *frequency-invariant* nonnegative vector. We refer to this model as the weight-shared jointly-diagonalizable (WJD) full-rank

spatial model. Because $\tilde{\mathbf{g}}_n$ is estimated by taking all frequency bins into account, the permutation problem is expected to be mitigated, resulting in performance improvement from FastMNMF1.

Substituting Eq. (4.21) into Eq. (1.6), the log-likelihood function of the parameters $\mathbf{W}$, $\mathbf{H}$, $\tilde{\mathbf{G}} = \{\tilde{\mathbf{g}}_n\}_{n=1}^N$, and $\mathbf{Q}$ is given by

$$\begin{aligned}\log p(\mathbf{X}|\mathbf{W},\mathbf{H},\tilde{\mathbf{G}},\mathbf{Q}) &= \sum_{f,t=1}^{F,T} \log \mathcal{N}_{\mathbb{C}}\left(\mathbf{x}_{ft} \,\middle|\, \mathbf{0}, \sum_{n=1}^{N} \lambda_{ftn}\mathbf{Q}_f^{-1}\mathrm{Diag}(\tilde{\mathbf{g}}_n)\mathbf{Q}_f^{-\mathsf{H}}\right) \\ &= -\sum_{f,t,m=1}^{F,T,M}\left(\frac{\tilde{x}_{ftm}}{\tilde{y}_{ftm}} + \log \tilde{y}_{ftm}\right) + T\sum_{f=1}^{F}\log\left|\mathbf{Q}_f\mathbf{Q}_f^{\mathsf{H}}\right| + \mathrm{const},\end{aligned} \tag{4.22}$$

where $\tilde{x}_{ftm} \triangleq |\mathbf{q}_{fm}^{\mathsf{H}}\mathbf{x}_{ft}|^2$ and $\tilde{y}_{ftm} \triangleq \sum_{n,k} w_{nkf}h_{nkt}\tilde{g}_{nm}$.

To avoid the scale ambiguity, we put the normalization constraints given by Eq. (4.6) and

$$\sum_{m=1}^{M} \tilde{g}_{nm} = 1, \tag{4.23}$$

$$\mathrm{tr}\left(\mathbf{Q}_f\mathbf{Q}_f^{\mathsf{H}}\right) = M. \tag{4.24}$$

FastMNMF2 is a special case of FastMNMF1, and ILRMA is a special case of FastMNMF2. The numbers of parameters of MNMF, FastMNMF1, FastMNMF2, and ILRMA for SCMs are $FNM(M+1)/2$, FM^2+FNM, FM^2+NM, and FM^2, respectively. The computational times, convergence speeds, and performances of these methods with different values of N, M, K, and F are evaluated in Sections 4.7.3 and 4.7.4.

4.3.2 Optimization

The parameters $\mathbf{W}$, $\mathbf{H}$, $\tilde{\mathbf{G}}$, and $\mathbf{Q}$ are updated in the same way as FastMNMF1. The MU update rules for $\mathbf{W}$, $\mathbf{H}$, and $\tilde{\mathbf{G}}$ are given by

$$w_{nkf} \leftarrow w_{nkf}\sqrt{\frac{\sum_{t,m=1}^{T,M} h_{nkt}\tilde{g}_{nm}\tilde{x}_{ftm}\tilde{y}_{ftm}^{-2}}{\sum_{t,m=1}^{T,M} h_{nkt}\tilde{g}_{nm}\tilde{y}_{ftm}^{-1}}}, \tag{4.25}$$

$$h_{nkt} \leftarrow h_{nkt} \sqrt{\frac{\sum_{f,m=1}^{F,M} w_{nkf} \tilde{g}_{nm} \tilde{x}_{ftm} \tilde{y}_{ftm}^{-2}}{\sum_{f,m=1}^{F,M} w_{nkf} \tilde{g}_{nm} \tilde{y}_{ftm}^{-1}}}, \tag{4.26}$$

$$\tilde{g}_{nm} \leftarrow \tilde{g}_{nm} \sqrt{\frac{\sum_{f,t,k=1}^{F,T,K} w_{nkf} h_{nkt} \tilde{x}_{ftm} \tilde{y}_{ftm}^{-2}}{\sum_{f,t,k=1}^{F,T,K} w_{nkf} h_{nkt} \tilde{y}_{ftm}^{-1}}}. \tag{4.27}$$

$\mathbf{Q}_f$ is updated in the same way as FastMNMF1 using Eq. (4.11). To satisfy the normalization constraints given by Eqs. (4.6), (4.23), and (4.24), we adjust the scales of $\mathbf{Q}$, $\tilde{\mathbf{G}}$, and $\mathbf{W}$ in this order in each iteration by using

$$\mu_f \triangleq \frac{1}{M} \mathrm{tr}(\mathbf{Q}_f \mathbf{Q}_f^{\mathsf{H}}), \quad \begin{cases} \mathbf{Q}_f \leftarrow \mu_f^{-\frac{1}{2}} \mathbf{Q}_f, \\ w_{nkf} \leftarrow \mu_f^{-1} w_{nkf}, \end{cases} \tag{4.28}$$

$$\phi_n \triangleq \sum_{m=1}^{M} \tilde{g}_{nm}, \qquad \begin{cases} \tilde{g}_{nm} \leftarrow \phi_n^{-1} \tilde{g}_{nm}, \\ w_{nkf} \leftarrow \phi_n w_{nkf}, \end{cases} \tag{4.29}$$

and Eq. (4.15).

4.3.3 Connection to Direction-Aware MNMF

FastMNMF2 has a connection to direction-aware MNMF [81, 82, 146] based on a factorizable full-rank spatial model given by

$$\mathbf{G}_{nf} = \sum_{d=1}^{D} z_{nd} \mathbf{R}_{fd}, \tag{4.30}$$

where D is the number of possible directions taken into account, $\mathbf{R}_{fd}$ is the SCM of direction d at frequency f, and z_{nd} is the weight of direction d for source n. Similarly to Eq. (4.21), Eq. (4.30) represents $\mathbf{G}_{nf}$ as the weighted sum of basis SCMs, and the weights are shared over all frequency bins. A difference is that direction-aware MNMF can be used only under a non-blind condition; only the magnitude part of $\mathbf{R}_{fd}$ is estimated, while the phase part of $\mathbf{R}_{fd}$ is fixed to that of the geometrically-computed SCM of direction d at frequency f. This method thus tends to fail in an unseen acoustic environment.

4.4 FastMNMF with a Deep Speech prior (FastMNMF-DSP)

FastMNMF-DSP can be derived by replacing NMF-based source model for one of sources with DNN-based speech model as in MNMF-DSP. Here we mainly discuss FastMNMF1-DSP because FastMNMF2-DSP can be derived in almost the same way.

4.4.1 Formulation

We assume that the observed signals include only one speech and N' ($\triangleq N-1$) noise (N sources in total), and source 0 corresponds to speech and source $n\,(1 \leq n \leq N')$ corresponds to noise. Now $\{\lambda_{nft}\}_{n=0}^{N'}$ are given as follows:

$$\lambda_{nft} = \begin{cases} u_f v_t [\boldsymbol{\sigma}_{\boldsymbol{\theta}}^2(\mathbf{z}_t)]_f & (n=0), \\ \displaystyle\sum_{k=1}^{K} w_{nkf} h_{nkt} & (1 \leq n \leq N'). \end{cases} \tag{4.31}$$

The log-likelihood function of FastMNMF1-DSP $\log p(\mathbf{X}|\mathbf{W},\mathbf{H},\mathbf{U},\mathbf{V},\mathbf{Z},\tilde{\mathbf{G}},\mathbf{Q})$ is obtained by substituting $\tilde{y}_{ftm} = u_f v_t [\boldsymbol{\sigma}_{\boldsymbol{\theta}}^2(\mathbf{z}_t)]_f \tilde{g}_{0fm} + \sum_{n=1}^{N'} \sum_{k=1}^{K} w_{nkf} h_{nkt} \tilde{g}_{nfm}$ into Eq. (4.3). To avoid the scale ambiguity, we put the normalization constraints given by Eqs. (4.4), (4.5), and (4.6) and

$$\sum_{f=1}^{F} u_f = 1 \tag{4.32}$$

For FastMNMF2-DSP, the normalization constraints are given by Eqs. (4.6), (4.23), (4.24), and (4.32).

4.4.2 Optimization

To update the latent variables $\mathbf{Z}$ included in Eq. (4.31), we use Metropolis sampling. A proposal $\mathbf{z}_t^{\text{new}} \sim \mathcal{N}(\mathbf{z}_t^{\text{old}}, \epsilon\mathbf{I})$ is accepted with probability $\min(1, \gamma_t)$, where γ_t is given by

$$\log \gamma_t = -\sum_{f,m=1}^{F,M} \left(\frac{\tilde{x}_{ftm}}{u_f v_t [\boldsymbol{\sigma}_{\boldsymbol{\theta}}^2(\mathbf{z}_t^{\text{new}})]_f \tilde{g}_{0fm} + \tilde{y}_{ftm}^{\text{noise}}} - \frac{\tilde{x}_{ftm}}{u_f v_t [\boldsymbol{\sigma}_{\boldsymbol{\theta}}^2(\mathbf{z}_t^{\text{old}})]_f \tilde{g}_{0fm} + \tilde{y}_{ftm}^{\text{noise}}} \right)$$

$$-\sum_{f,m=1}^{F,M}\log\frac{u_f v_t[\boldsymbol{\sigma}_{\boldsymbol{\theta}}^2(\mathbf{z}_t^{\text{new}})]_f\tilde{g}_{0fm}+\tilde{y}_{ftm}^{\text{noise}}}{u_f v_t[\boldsymbol{\sigma}_{\boldsymbol{\theta}}^2(\mathbf{z}_t^{\text{old}})]_f\tilde{g}_{0fm}+\tilde{y}_{ftm}^{\text{noise}}}, \tag{4.33}$$

where $\tilde{y}_{ftm}^{\text{noise}}\stackrel{\text{def}}{=}\sum_{n=1}^{N'}\lambda_{nft}\tilde{g}_{nfm}$. As in the NMF-based source model, the MU rules of $\mathbf{U}$ and $\mathbf{V}$ are given by

$$u_f\leftarrow u_f\sqrt{\frac{\sum_{t,m=1}^{T,M}v_t[\boldsymbol{\sigma}_{\boldsymbol{\theta}}^2(\mathbf{z}_t)]_f\tilde{g}_{0fm}\tilde{x}_{ftm}\tilde{y}_{ftm}^{-2}}{\sum_{t,m=1}^{T,M}v_t[\boldsymbol{\sigma}_{\boldsymbol{\theta}}^2(\mathbf{z}_t)]_f\tilde{g}_{0fm}\tilde{y}_{ftm}^{-1}}}, \tag{4.34}$$

$$v_t\leftarrow v_t\sqrt{\frac{\sum_{f,m=1}^{F,M}u_f[\boldsymbol{\sigma}_{\boldsymbol{\theta}}^2(\mathbf{z}_t)]_f\tilde{g}_{0fm}\tilde{x}_{ftm}\tilde{y}_{ftm}^{-2}}{\sum_{f,m=1}^{F,M}u_f[\boldsymbol{\sigma}_{\boldsymbol{\theta}}^2(\mathbf{z}_t)]_f\tilde{g}_{0fm}\tilde{y}_{ftm}^{-1}}}. \tag{4.35}$$

$\mathbf{W}$, $\mathbf{H}$, $\tilde{\mathbf{G}}$, and $\mathbf{Q}$ are updated in the same way as FastMNMF1 using Eqs. (4.7), (4.8), (4.9), and (4.11).

To satisfy the normalization constraints given by Eqs. (4.4), (4.5), (4.6), and (4.32), we adjust the scales of $\mathbf{Q}$, $\tilde{\mathbf{G}}$, $\mathbf{U}$, and $\mathbf{W}$ in this order in each iteration by using

$$\mu_{fm}\triangleq\mathbf{q}_{fm}^{\mathsf{H}}\mathbf{q}_{fm},\quad\begin{cases}\mathbf{q}_{fm}\leftarrow\mu_{fm}^{-\frac{1}{2}}\mathbf{q}_{fm},\\ \tilde{g}_{nfm}\leftarrow\mu_{fm}^{-1}\tilde{g}_{nfm}\end{cases} \tag{4.36}$$

$$\phi_{nf}\triangleq\sum_{m=1}^{M}\tilde{g}_{nfm},\quad\begin{cases}\tilde{g}_{nfm}\leftarrow\phi_{nf}^{-1}\tilde{g}_{nfm},\\ u_f\leftarrow\phi_{1f}u_f,\\ w_{nkf}\leftarrow\phi_{nf}w_{nkf}\quad(1\le n\le N'),\end{cases} \tag{4.37}$$

$$\psi\triangleq\sum_{f=1}^{F}u_f,\quad\begin{cases}u_f\leftarrow\psi^{-1}u_f\\ v_t\leftarrow\psi v_t,\end{cases} \tag{4.38}$$

and Eq. (4.15). In FastMNMF2-DSP, we adjust the scale by using

$$\mu_f\triangleq\frac{1}{M}\mathrm{tr}(\mathbf{Q}_f\mathbf{Q}_f^{\mathsf{H}}),\quad\begin{cases}\mathbf{Q}_f\leftarrow\mu_f^{-\frac{1}{2}}\mathbf{Q}_f,\\ u_f\leftarrow\mu_f^{-1}u_f,\\ w_{nkf}\leftarrow\mu_f^{-1}w_{nkf},\end{cases} \tag{4.39}$$

$$\phi_n\triangleq\sum_{m=1}^{M}\tilde{g}_{nm},\quad\begin{cases}\tilde{g}_{nm}\leftarrow\phi_n^{-1}\tilde{g}_{nm},\\ u_f\leftarrow\phi_1 u_f,\\ w_{nkf}\leftarrow\phi_n w_{nkf}\quad(1\le n\le N'),\end{cases} \tag{4.40}$$

and Eqs. (4.38) and (4.15).

4.5 Initialization

We explain four parameter initialization methods for FastMNMF, *i.e.*, random, diagonal, circular, and gradual initialization methods. The parameters $\mathbf{W}$ and $\mathbf{H}$ of the low-rank source model are initialized randomly and the parameters $\tilde{\mathbf{G}}$ and $\mathbf{Q}$ of the JD full-rank spatial model are initialized with one of the four methods. As experimentally shown in Section 4.7.5, the gradual initialization method works best in practice. We here consider an (over)determined condition ($N \leq M$), which is considered to be practically important, and discuss only FastMNMF1 because FastMNMF2 can be initialized in the same way.

4.5.1 Random Initialization

In the random initialization method, the diagonalizer $\mathbf{Q}_f$ is initialized to an identity matrix and $\tilde{\mathbf{g}}_{nf}$ is initialized randomly. Although FastMNMF is considered to be less sensitive to the initialization than MNMF because of the restricted model complexity, FastMNMF is still more likely to get stuck in bad local optima than ILRMA (constrained version of FastMNMF), when the random initialization method is used.

4.5.2 Diagonal Initialization

Inspired by the relation between FastMNMF1 and ILRMA discussed in Section 4.2.4, in the diagonal initialization method, $\mathbf{Q}_f$ is initialized to an identity matrix and $\tilde{\mathbf{g}}_{:f} \in \mathbb{R}^{N \times M}$ is initialized to a pseudo-diagonal matrix as follows:

$$\tilde{\mathbf{g}}_{:f} = \begin{pmatrix} 1 & \epsilon & \dots & \epsilon & \epsilon & \epsilon & \dots \\ \epsilon & 1 & \dots & \epsilon & \epsilon & \epsilon & \dots \\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \ddots \\ \epsilon & \epsilon & \dots & 1 & \epsilon & \epsilon & \dots \end{pmatrix}, \tag{4.41}$$

where ':' indicates a set of all indices and ϵ is a small number (*e.g.*, $\epsilon = 10^{-2}$). Under a determined condition ($N = M$), $\tilde{\mathbf{g}}_{:f}$ is a square matrix close to an identity matrix. Although $\tilde{\mathbf{g}}_{nf}$ is updated iteratively, FastMNMF1 starting with $\tilde{\mathbf{g}}_{nf} \approx \mathbf{e}_n$ is expected to work as stably as ILRMA with $\tilde{\mathbf{g}}_{nf} = \mathbf{e}_n$. Under an overdetermined

condition ($N < M$), however, the pseudo-demixing filters $\{\mathbf{q}_{fm}\}_{m=N+1}^{M}$ work ineffectively in the early iterations because the weights $\{\tilde{g}_{nfm}\}_{m=N+1}^{M}$ of each source n are small, *i.e.*, at most only N possible directions can be considered for N sources. In fact, we found that overdetermined FastMNMF1 with M microphones is comparable with determined FastMNMF1 using only the first N microphones, when the diagonal initialization method is used.

4.5.3 Circular Initialization

To solve the potential problem of the diagonal initialization under an overdetermined condition, in the circular initialization method, $\mathbf{Q}_f$ is set to an identity matrix and $\tilde{\mathbf{g}}_{:f} \in \mathbb{R}^{N\times M}$ is set to a pseudo-circulant matrix as follows:

$$\tilde{\mathbf{g}}_{:f} = \begin{pmatrix} 1 & \epsilon & \dots & \epsilon & 1 & \epsilon & \dots \\ \epsilon & 1 & \dots & \epsilon & \epsilon & 1 & \dots \\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \ddots \\ \epsilon & \epsilon & \dots & 1 & \epsilon & \epsilon & \dots \end{pmatrix}. \tag{4.42}$$

In this case, M possible directions are considered for N sources even in the early iterations.

4.5.4 Gradual Initialization

We define the gradual initialization method as follows: Inspired by the stable behavior of ILRMA with $K = 2$, FastMNMF1 with $K = 2$ is initialized by the circular initialization method. After updating $\mathbf{W}$, $\mathbf{H}$, $\mathbf{Q}$, and $\tilde{\mathbf{G}}$ 50 times, K is increased to a larger number and only $\mathbf{W}$ and $\mathbf{H}$ are reset to random values. This method was found to work stably among several possible implementations of gradual initialization.

In an overdetermined case ($N_{\text{true}} < M$), for example, another option is to first use determined FastMNMF1 ($N = M$) with $K = 2$ assuming the presence of $N - N_{\text{true}}$ extra sources (noise or reverberation), where N_{true} represents the number of true sources. After updating $\mathbf{W}$, $\mathbf{H}$, $\mathbf{Q}$, and $\tilde{\mathbf{G}}$ 50 times, K is set to a larger number and $\mathbf{W}$ and $\mathbf{H}$ are reset to random values. To obtain $\tilde{\mathbf{g}}_{:f} \in \mathbb{R}^{N_{\text{true}}\times M}$, one needs to select N_{true} rows from the estimated $\tilde{\mathbf{g}}_{:f} \in \mathbb{R}^{N\times M}$. Although this

could be done as in rank-constrained FastMNMF (Section 4.6.1), wrong selection degrades the separation performance. This option was thus not used in our experiment.

4.6 FastMNMF with a Rank-Constrained Spatial Model (RC-FastMNMF)

We propose rank-constrained FastMNMF, a special case of FastMNMF that enables us to explicitly specify the rank of the SCMs $\{\mathbf{G}_{nf}\}_{f=1}^{F}$ of each source n according to its directivity. We here discuss rank-constrained FastMNMF1 because rank-constrained FastMNMF2 can be derived straightforwardly. As discussed in Section 4.2.4, the number of non-zero elements of $\tilde{\mathbf{g}}_{nf}$ indicates the rank of $\mathbf{G}_{nf}$ in the JD full-rank spatial model. In the MU rule given by Eq. (4.9), once some elements of $\tilde{\mathbf{g}}_{nf}$ are set to zero, they are kept to zero. Rank-constrained FastMNMF1 can thus be obtained by initializing a specified number of elements of $\tilde{\mathbf{g}}_{nf}$ to zero, where the dimensions of those elements should be selected carefully for each source n according to the surrounding acoustic environment. Typically, $\tilde{\mathbf{g}}_{nf}$ is initialized to a one-hot or all-one vector for a directional or diffuse sound, respectively. We explain how to initialize rank-constrained FastMNMF1 for source enhancement or separation, where one or more directional sources (*e.g.*, speakers) exist with diffuse noise, respectively.

4.6.1 Source Separation

Suppose that there are L directional sources (target) and $N - L$ diffuse sources (noise). First, $\mathbf{Q}$ is initialized with the gradual initialization method described in Section 4.5.4. Since $\mathbf{Q}_f$ is a pseudo-demixing matrix at frequency f, the spectrogram of component m is given by $\{\hat{x}_{ftm} \triangleq \mathbf{q}_{fm}^{\mathsf{H}}\mathbf{x}_{ft}\}_{f,t=1}^{F,T}$. Using the projection-back method [39] for solving the scale ambiguity of each component, the image of component m is given by $\{u_{fmm'}\hat{x}_{ftm}\}_{f,t,m'=1}^{F,T,M}$, where $\mathbf{u}_{fm}$ (column vectors of $\mathbf{Q}_f^{-1} = \mathbf{U}_f$) is a pseudo-steering vector of component m at frequency f.

Let v_m be the maximum frame-wise power of component m, *i.e.*,

$$v_m \triangleq \max_t \sum_{f,m'=1}^{F,M} |u_{fmm'} \mathbf{q}_{fm}^{\mathsf{H}} \mathbf{x}_{ft}|^2. \tag{4.43}$$

The component indices $(1 \leq m \leq M)$ are then sorted in a descending order with respect to the significance $\{v_m\}_{m=1}^M$ and the M rows of $\mathbf{Q}_f$ are permuted accordingly. Assuming that the top L components correspond to the target $(L < N \leq M)$, $\{\tilde{\mathbf{g}}_{nf}\}_{n=1}^L$ and $\{\tilde{\mathbf{g}}_{nf}\}_{n=L+1}^N$ are initialized to one-hot and all-one vectors, respectively, as follows:

$$\tilde{\mathbf{g}}_{:f} = \begin{pmatrix} 1 & 0 & \dots & 0 & \dots & 0 \\ 0 & 1 & \dots & 0 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\ 0 & 0 & \dots & 1 & \dots & 0 \\ 1 & 1 & \dots & 1 & \dots & 1 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 1 & 1 & \dots & 1 & \dots & 1 \end{pmatrix}. \tag{4.44}$$

4.6.2 Source Enhancement

Source enhancement is equivalent to source separation with $L = 1$. Assuming that the target source is predominant in the mixture, the pseudo-steering vectors $\mathbf{u}_{f1}$ and $\{\mathbf{u}_{fm}\}_{m=2}^M$ are initialized to the most principal and the remaining eigenvectors of the empirical SCM given by $\sum_t \mathbf{X}_{ft}$, respectively. Then, $\mathbf{Q}$ is initialized with the gradual initialization method described in Section 4.5.4. The subsequent part is the same as source separation.

The key feature of rank-constrained FastMNMF is that the rank-1 SCMs of L directional target sources and the full-rank SCMs of $N - L$ diffuse noise sources are estimated jointly, where N is an arbitrary number and the noise sources are assumed to exist on M directions including the target directions. When ILRMA is used, in contrast, M directional sources are assumed to *exclusively* exist on M directions, resulting in L rank-1 target SCMs and a rank-$(M - L)$ noise SCM. An additional step is thus required for recovering the full-rank noise SCM [79]. The superiority of rank-constrained FastMNMF in speech enhancement and source separation is experimentally validated in Sections 4.7.7 and 4.7.8.

4.7 Evaluation

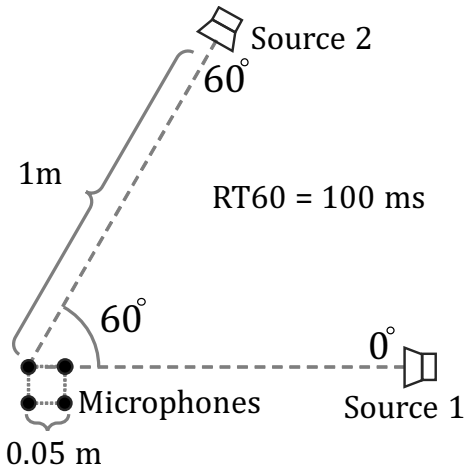
This section reports comparative experiments conducted for evaluating the effectiveness of FastMNMF. First, we investigated the interpretation of the joint diagonalization constraint described in Section 4.2.4. Second, we compared the separation performances and computational efficiencies of FastMNMF, ILRMA [28], and MNMF [27] for speech separation while the theoretical complexities of these methods are given in Section 4.3.1. To draw the full potential of FastMNMF, we comprehensively investigated the configuration of N, M, K, and F and compared the four initialization methods described in Section 4.5. Finally, we tested rank-constrained FastMNMF for speech enhancement and separation as described in Section 4.6. Through all experiments, audio signals were sampled at 16 kHz and processed by STFT with a shifting interval of 512 points and a Hann window of 2048 points ($F = 1025$), unless otherwise noted.

4.7.1 Validation of Directivity Awareness

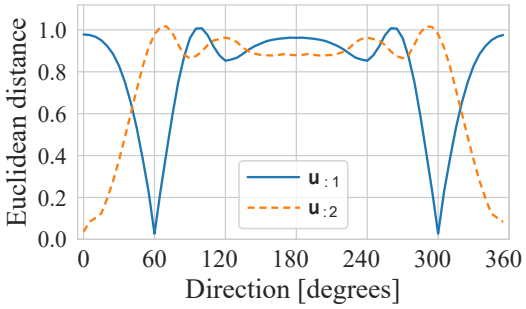
We validated our hypothesis that $\tilde{\mathbf{g}}_{nf}$ indicates the weights of M directions for source n. If this is true, some column vectors of $\mathbf{U}_f = \mathbf{Q}_f^{-1}$ estimated by FastMNMF would coincide with the steering vectors of source directions because $\mathbf{U}_f$ can be regarded as a pseudo-mixing matrix.

Experimental Conditions

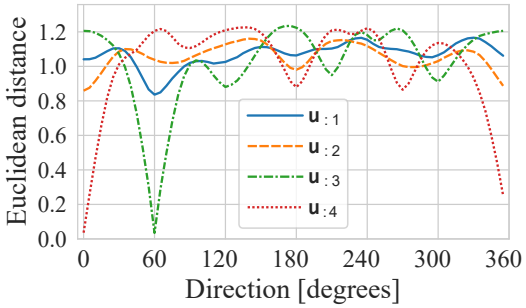
We investigated a determined case ($N = M = 2$) and an overdetermined case ($N = 2, M = 4$), where the sources and microphones were located as depicted in Fig. 4.4(a) and only the upper two microphones were used in the determined case. Using *Pyroomacoustics* library [147], we simulated the steering vectors $\{\mathbf{a}_{fd} \in \mathbb{C}^M\}_{d=1}^D$ of equally-spanned directions (azimuths; $D = 72$) with the reverberation time of $\mathrm{RT}_{60} = 100$ ms. We made an M-channel mixture signal of 6.9 seconds by spatially mixing two speech signals ($N = 2$) randomly selected from the WSJ-0 corpus [140] with the steering vectors $\mathbf{a}_{f1}$ and $\mathbf{a}_{f13}$ corresponding to the source directions (0 and 60 degrees).



(a) Recording condition.



(b) $N = M = 2$.



(c) $M = 4$ and $N = 2$.

Figure 4.4: The Euclidean distances between the estimated pseudo-steering vectors $\{\mathbf{u}_{fm}\}_{m=1}^{M}$ of M directions and the ground-truth steering vectors $\{\mathbf{a}_{fd}\}_{d=1}^{D}$ of all possible D directions accumulated over all frequency bins.

FastMNMF1 with the circular initialization method and $K = 16$ was used for estimating the pseudo-demixing matrix $\mathbf{Q}_f = \mathbf{U}_f^{-1}$. The Euclidean distances between each pseudo-steering vector $\mathbf{u}_{fm}$ (the m-th column vector of $\mathbf{U}_f$) and the true steering vectors $\{\mathbf{a}_{fd}\}_{d=1}^{D}$ were computed at each frequency f and the average distance was computed over all frequency bins. Note that all vectors were normalized in advance such that the L2 norm of each vector is equal to 1 and the phase of the first channel was equal to zero.

Experimental Results

As shown in Fig. 4.4(b), when $N = M = 2$, we observed the expected correspondence, *i.e.*, the estimated result of $\mathbf{u}_{f1}$ was closest to $\mathbf{a}_{f13}$ (60 degree) and that of $\mathbf{u}_{f2}$ was closest to $\mathbf{a}_{f1}$ (0 degrees). Note that the estimated result of $\mathbf{u}_{f1}$ was also closest to $\mathbf{a}_{f61}$ (300 degrees) because of the front-back ambiguity of the

straight-shape microphone array. We also confirmed that the estimated result of $\tilde{\mathbf{g}}_{1f}$ was close to $[1, \epsilon]^\mathsf{T}$ and that of $\tilde{\mathbf{g}}_{2f}$ was close to $[\epsilon, 1]^\mathsf{T}$, where ϵ indicates a small value. As shown in Fig. 4.4(c), when $N = 2$ and $M = 4$, we found the estimated result of $\mathbf{u}_{f3}$ was closest to $\mathbf{a}_{f13}$ (60 degrees) and that of $\mathbf{u}_{f4}$ was closest to $\mathbf{a}_{f1}$ (0 degree), and those of $\mathbf{u}_{f1}$ and $\mathbf{u}_{f2}$, which did not correspond to any of $\{\mathbf{a}_{fd}\}_{d=1}^{D}$, were considered to represent the reverberation of the two sources. In fact, the estimated result of $\tilde{\mathbf{g}}_{1f}$ was close to $[\epsilon, \epsilon, 1, \epsilon]^\mathsf{T}$ and that of $\tilde{\mathbf{g}}_{2f}$ was close to $[\epsilon, \epsilon, \epsilon, 1]^\mathsf{T}$, and they indicated the large weights of the two clear directions (the third and fourth dimensions corresponding to 60 and 0 degrees) and the small weights of the two vague directions (the first and second dimensions corresponding to the reverberation). This result clearly supports our hypothesis on the directivity awareness of FastMNMF1 and justify the inter-frequency weight sharing of FastMNMF2.

4.7.2 Basic Configurations for Speech Separation

We compared the separation performances and computational efficiencies of FastMNMF, ILRMA [28], and MNMF [27] in a speech separation task. We randomly selected 100 simulated echoic three-speaker mixture signals ($N_{\text{true}} = 3, M = 8$) from the evaluation dataset of spatialized WSJ0-mix [144], where the positions of sources and microphones had been randomly determined for each mixture. The average SDR of the input mixture signals (the first channel) was -3.1 dB. FastMNMF and MNMF were directly tested with the overdetermined setting ($N = 3, M = 8$). In addition, all methods were tested with the determined setting ($N = M = 8$). For evaluation, N_{true} sources were selected from N estimated sources in a retrospective manner such that the SDR was maximized. Although this strategy was advantageous for the determined setting, we aimed to eliminate the impact of an arbitrary selection method and show the maximum potential of determined BSS methods including ILRMA. For the low-rank source model, $K \in \{2, 4, 16, 64, 256\}$ was used, and $\mathbf{W}$ and $\mathbf{H}$ were initialized randomly.

FastMNMF was initialized with the circular or gradual initialization methods (Sections 4.5.3 and 4.5.4). For ILRMA, the demixing matrices $\mathbf{D}$ were initialized

Table 4.1: Elapsed times [sec] per iteration for processing 8ch signals of 10 [sec] on CPU (Intel Xeon W-2145 3.70 GHz).

Method		ILRMA					FastMNMF1					FastMNMF2					MNMF				
# of bases K		2	4	16	64	256	2	4	16	64	256	2	4	16	64	256	2	4	16	64	256
# of sources N	3	-	-	-	-	-	1.61	1.60	1.69	2.02	3.47	1.58	1.60	1.65	1.99	3.39	21.7	21.7	21.8	22.1	23.7
	8	1.41	1.43	1.56	2.22	4.87	2.09	2.13	2.35	3.23	6.89	2.04	2.11	2.31	3.18	6.90	30.0	30.1	30.3	31.1	35.0

to identity matrices. Alternatively, ILRMA was initialized with the gradual initialization method because it was a special case of FastMNMF. For MNMF, the SCMs $\mathbf{G}$ were initialized to identity matrices. Alternatively, MNMF was initialized with ILRMA, *i.e.*, the SCM of each dominant source n $(1 \leq n \leq N_{\text{true}})$ was given by $\mathbf{G}_{nf} = \mathbf{a}_{nf}\mathbf{a}_{nf}^{\mathsf{H}} + \epsilon\mathbf{I}$, where $\mathbf{a}_{nf}$ is a steering vector such that $\mathbf{A}_f \triangleq [\mathbf{a}_{1f}, \ldots, \mathbf{a}_{Nf}] \triangleq \mathbf{D}_f^{-1}$ and $\epsilon = 10^{-2}$ is a small number to make $\mathbf{G}_f$ a full-rank matrix. In all methods, the total number of iterations (including 50 iterations for initial FastMNMF with $K = 2$ in the gradual initialization method or 50 iterations for ILRMA in the initialization of MNMF) was set to 200.

The signal-to-distortion ratio (SDR) [136, 137] was used for evaluating the separation performance. To investigate the convergence speed of each method, the SDR evolution was monitored. The elapsed time per iteration for processing a 10-s mixture signal was measured on Intel Xeon W-2145 (3.70 GHz).

4.7.3 Comparison of FastMNMF with ILRMA and MNMF

Table 4.1 lists the elapsed times per iteration. There was no significant difference between FastMNMF1 and FastMNMF2. FastMNMF was more than 10 or 5 times faster than MNMF for $K = 2$ or $K = 256$, respectively. An interesting finding was that ILRMA with $N = 8$ was 1.5 times faster than FastMNMF with $N = 8$ regardless of K, but was almost as fast as FastMNMF with $N = 3$. Especially, FastMNMF with $N = 3$ with larger K tended to be faster than ILRMA. This indicates the effectiveness of considering only N_{true} sources for saving the computational cost under an overdetermined condition ($N < M$).

Fig. 4.5 shows the SDR evolutions of FastMNMF, ILRMA, and MNMF averaged over the 100 mixtures, and Fig. 4.6 shows the evolutions of the log-likelihoods, which is the objective function for the parameter estimation. In FastMNMF

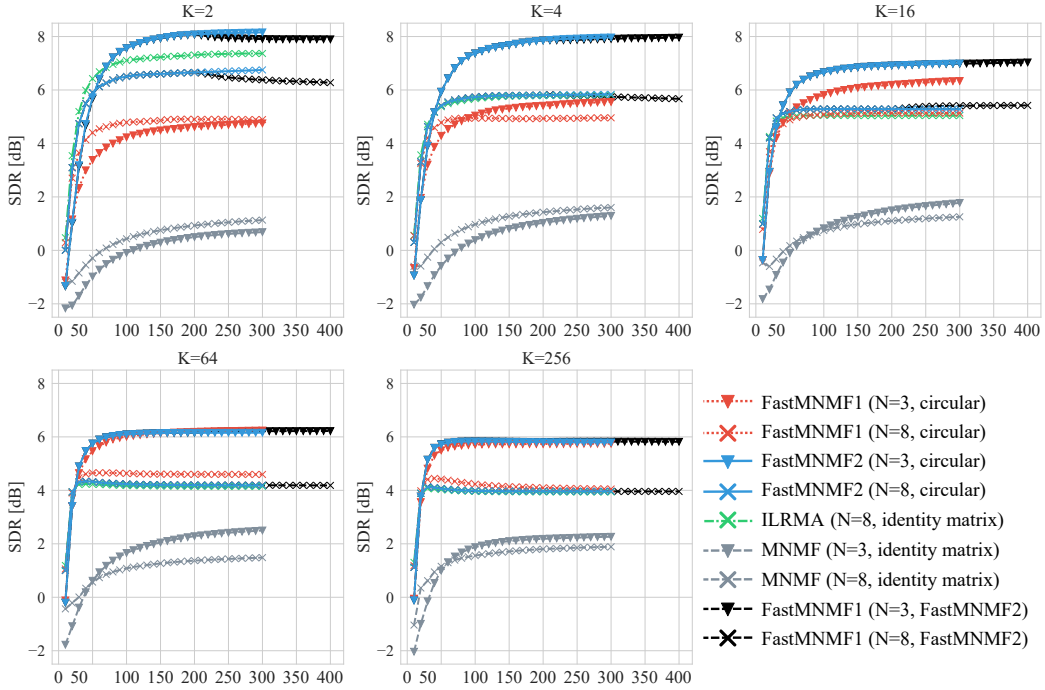


Figure 4.5: The evolutions of average SDRs [dB]. Crosses and triangles indicate the determined and overdertermined settings, respectively.

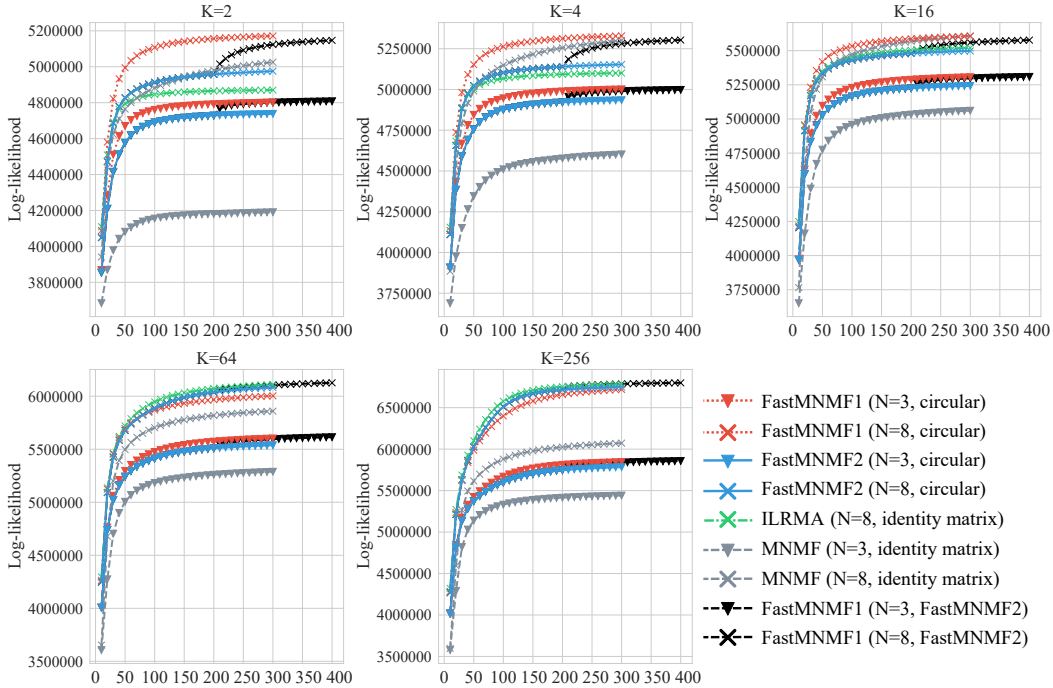


Figure 4.6: The evolutions of average log-likelihoods. Crosses and triangles indicate the determined and overdertermined settings, respectively.

and ILRMA with larger K, which did not necessarily work better, the SDRs converged faster, whereas the log-likelihoods converged slower. In ILRMA, among all the parameters, only the time-invariant demixing matrices were used for source separation based on Wiener filtering. The faster SDR convergence indicates that the demixing matrices got stuck at local optima within several tens of iterations, and the slower log-likelihood convergence indicates that only the NMF parameters $\mathbf{W}$ and $\mathbf{H}$ were updated continuously. In FastMNMF, although Wiener filtering was affected by all the parameters, the diagonalizers $\mathbf{Q}$, which had a similar role as the demixing matrices of ILRMA, are considered to be particularly important for separation performance. The convergence results indicate that $\mathbf{Q}$ got stuck at local optima quickly, and $\mathbf{W}$, $\mathbf{H}$, and $\tilde{\mathbf{G}}$ were updated continuously, as observed in ILRMA. Comparing the SDRs and log-likelihoods of the determined and overdetermined configurations of FastMNMF, the same problem occurred in the determined configurations. A possible reason why the demixing filters and the diagonalizers $\mathbf{Q}$ easily got stuck at local optima was that $\tilde{y}_{ftm} \triangleq \sum_{n,k} w_{nkf} h_{nkt} \tilde{g}_{nm}$ tended to overfit $\tilde{x}_{ftm} \triangleq |\mathbf{q}_{fm}^{\mathsf{H}} \mathbf{x}_{ft}|^2$ before $\mathbf{Q}$ was sufficiently optimized because of the richer expressive power of NTF with larger K or N.

While FastMNMF1 achieved higher log-likelihoods than FastMNMF2, FastMNMF2 achieved better SDRs than FastMNMF1, especially for $K = 2, 4$. The higher log-likelihoods of FastMNMF1 were reasonable because in general, a probabilistic model with a higher degree of freedom can achieve a higher likelihood in maximum likelihood estimation. To investigate the characteristics of FastMNMF1, we fully optimized $\mathbf{W}$, $\mathbf{H}$, $\tilde{\mathbf{G}}$, and $\mathbf{Q}$ until convergence via FastMNMF2 with 200 iterations and then further optimized them via FastMNMF1 with 200 iterations As shown in Fig. 4.5 and Fig. 4.6, the log-likelihoods restarted to increase when FastMNMF1 switched to FastMNMF2 as expected theoretically, especially for smaller K. On the other hand, the SDRs started to decrease for $K = 2$, otherwise remained unchanged. This indicates that higher log-likelihoods did not always mean higher SDRs in FastMNMF1, probably because physically-inconsistent parameters can fit more precisely the observed data and give higher

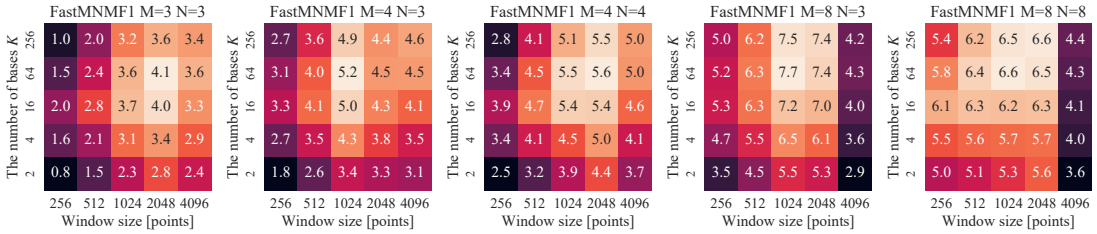
log-likelihoods. Note that the global optimum of FastMNMF1 is not always included in the parameter space of FastMNMF2. Nonetheless, the experimental fact that FastMNMF2 stably achieved better SDRs than FastMNMF1 indicates that the direction-weight sharing over all frequency bins effectively eliminates physically-inconsistent parameters from the solution space of FastMNMF2 and improved the consistency between SDRs and log-likelihoods.

In terms of the SDR, FastMNMF2 worked best for $K = 2, 4$, whereas FastMNMF1 worked best for $K = 16$. Note that the model complexities of FastMNMF1 and FastMNMF2 are relatively closer to those of MNMF and ILRMA, respectively (Fig. 4.1). Because even the excessively low-rank PSDs of the source model gave useful clues for estimating the constrained SCMs, FastMNMF2 and ILRMA worked better for smaller K. Because the precise estimate of source PSDs is required for estimating the SCMs with higher degrees of freedom, FastMNMF1 and MNMF tended to work better for larger K. The separation performance, however, was limited because of the insufficient optimization of $\mathbf{Q}$.

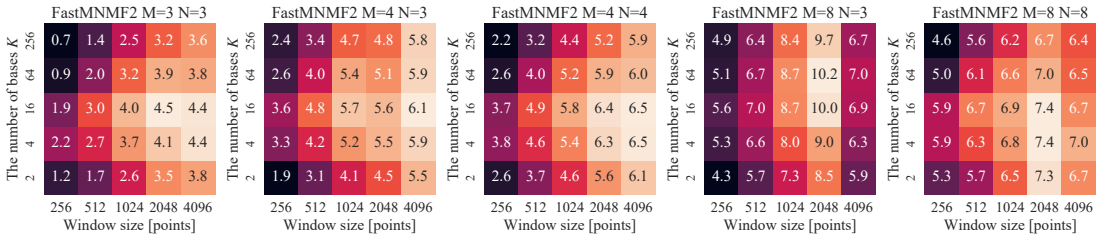
4.7.4 Comparison of Model Complexities for FastMNMF

In the speech separation task, we comprehensively investigated the SDRs and log-likelihoods obtained by FastMNMF with different complexities. Specifically, we tested FastMNMF with $N \in \{3, 4, 8\}$ sources, $M \in \{3, 4, 8\}$ microphones ($N = M$ or $N = N_{\text{true}} = 3$), $K \in \{2, 4, 16, 64, 256\}$ bases, and $F \in \{129, 257, 513, 1025, 2049\}$ bins, where STFT with a Hann window of $2(F-1)$ points and a shifting interval of $(F-1)/2$ points was used. In a determined setting, N_{true} sources were selected from N estimated sources for evaluation, as noted in Section 4.7.2. To draw the full potential of FastMNMF, it was initialized with the gradual initialization method described in Section 4.5.4.

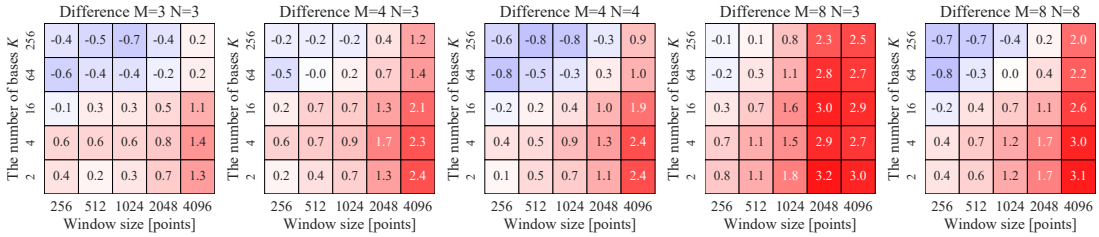
Figs. 4.7(a) and 4.7(b) show the SDRs of FastMNMF1 and FastMNMF2 averaged over the 100 mixtures, respectively, and Fig. 4.7(c) shows the gaps. FastMNMF2 with $N = 3$, $M = 8$, $K = 64$, and $F = 1025$ attained the best SDR (10.2 dB). For $K \in \{64, 256\}$, FastMNMF2 achieved higher log-likelihoods than FastMNMF1 in all cases except for $M = 8$ and $N = 3$. This means that FastMNMF1 got stuck



(a) SDRs obtained by FastMNMF1.



(b) SDRs obtained by FastMNMF2.

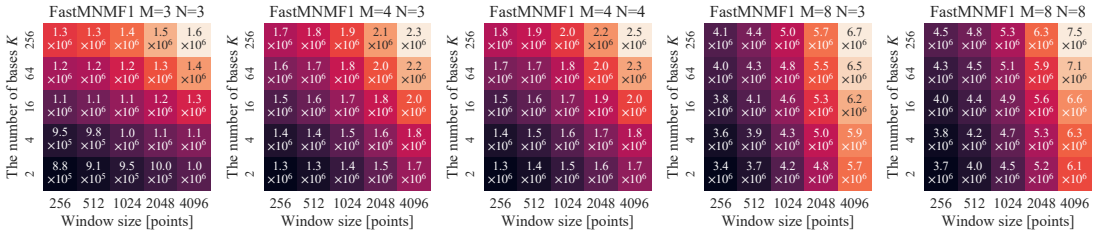


(c) SDR gaps between FastMNMF1 and FastMNMF2 (positive values indicate the superiority of FastMNMF2).

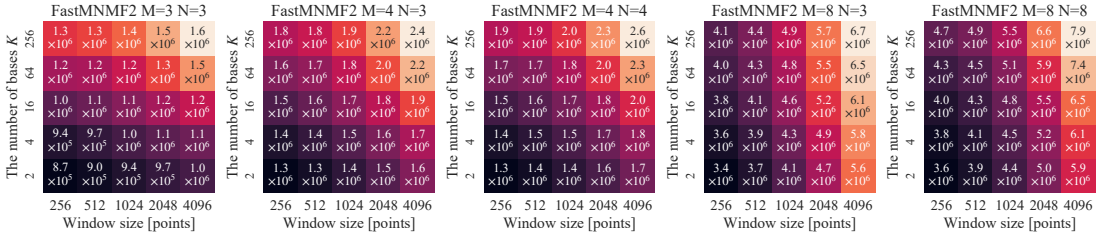
Figure 4.7: The average SDRs obtained by FastMNMF1 and FastMNMF2 with gradual initialization.

at local optima because the parameter space of FastMNMF1 includes that of FastMNMF2 and the log-likelihood at the global optimum of FastMNMF1 is the same as or better than that of FastMNMF2.

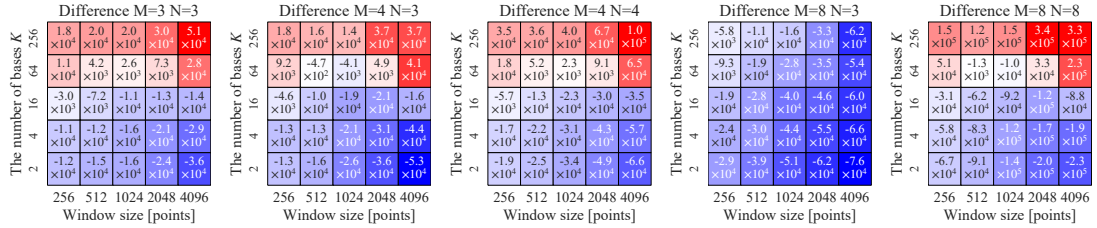
Figs. 4.8(a) and 4.8(b) show the log-likelihoods of FastMNMF1 and FastMNMF2 averaged over the 100 mixtures, respectively, and Fig. 4.7(c) shows the gaps. For $K \in \{2, 4, 16\}$, FastMNMF2 had lower log-likelihoods, but achieved higher SDRs than FastMNMF1 in almost all settings. For larger M and F, FastMNMF2 tended to outperform FastMNMF1 by a larger margin. Since the spatial models of FastMNMF1 and FastMNMF2 have $FM^2 + FNM$ and $FM^2 + NM$ parameters, respectively, the solution space of FastMNMF1 become increasingly wider than



(a) Log-likelihoods obtained by FastMNMF1.



(b) Log-likelihoods obtained by FastMNMF2.



(c) Log-likelihood gaps between FastMNMF1 and FastMNMF2 (positive values indicate the superiority of FastMNMF2).

Figure 4.8: The average log-likelihoods obtained by FastMNMF1 and FastMNMF2 with gradual initialization.

that of FastMNMF2 as M and F increase. This makes FastMNMF1 easy to get stuck at physically-inconsistent local optima, as discussed in Section 4.7.3.

4.7.5 Comparison of Initialization Methods for FastMNMF

In the speech separation task, we further investigated the SDRs obtained by FastMNMF with the random, diagonal, circular, and gradual initialization methods described in Section 4.5. The model complexities were set to $N = 3$, $M = 8$, $K \in \{2, 4, 16, 64, 256\}$, and $F = 1025$.

Fig. 4.9 shows the average SDRs over the 100 mixtures. FastMNMF2 with $K = 64$ and the gradual initialization method achieved the highest SDR (10.2

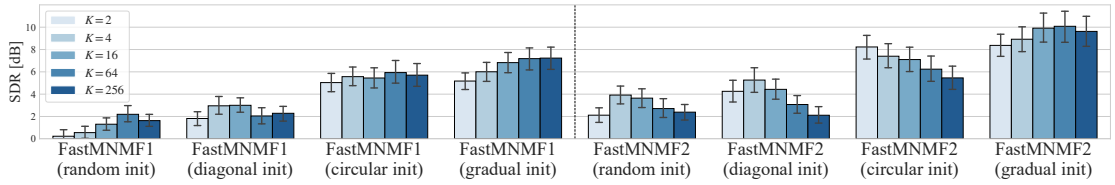


Figure 4.9: SDRs obtained by FastMNMF with the four initialization methods.

dB). For any K, FastMNMF1 with the circular initialization method significantly outperformed FastMNMF1 with the random or diagonal initialization method. The same can be said for FastMNMF2. While FastMNMF2 with the circular initialization method worked better for smaller K, FastMNMF2 with the gradual initialization method worked better for larger K. This indicates that FastMNMF2 with $K = 2$ was effectively used for mitigating the initialization sensitivity of FastMNMF2 with larger K in the gradual initialization method.

4.7.6 Comparison with State-of-the-Art BSS Methods

We compared the proposed FastMNMF with the IP method and another FastMNMF with the FPI method (FastMNMF1-FPI [44] and FastMNMF2-FPI). We also tested ILRMA with the component clustering mechanism [78], where ILRMA with $M = 8$ was used for estimating M components that were hardly clustered to $N = 4$ sources in advance ($P = 2$ components each). Moreover, we tested the soft-clustering version of [78] called two-step FastMNMF that fixes $\mathbf{Q}_f$ to $\mathbf{D}_f$ estimated with ILRMA and estimates only $\mathbf{W}$, $\mathbf{H}$, and $\tilde{\mathbf{G}}$.

Fig. 4.10 shows the average SDRs, SIRs, and SARs over the 100 mixtures. Because these three measures were consistent, we henceforth focus on the SDRs only. FastMNMF2 (10.2 dB) outperformed two-step FastMNMF1 (9.1 dB), two-step FastMNMF2 (7.8 dB), and ILRMA with the clustering mechanism (6.8 dB). This indicates that $\mathbf{Q}$ estimated by ILRMA was not optimal under the overdetermined condition and that the joint component separation and clustering was effective for improving the performance. A reason why two-step FastMNMF1 outperformed two-step FastMNMF2 was that when $\mathbf{Q}$ was fixed to one estimated by ILRMA, there was more room for performance improvement in FastMNMF1

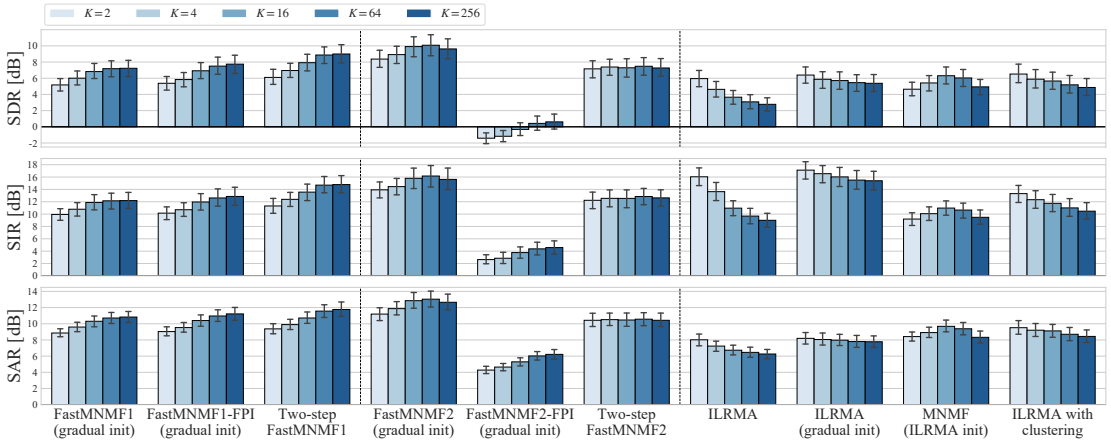


Figure 4.10: SDRs, SIRs, and SARs obtained by FastMNMF, FastMNMF-FPI, Two-step FastMNMF, ILRMA, and MNMF in speech separation.

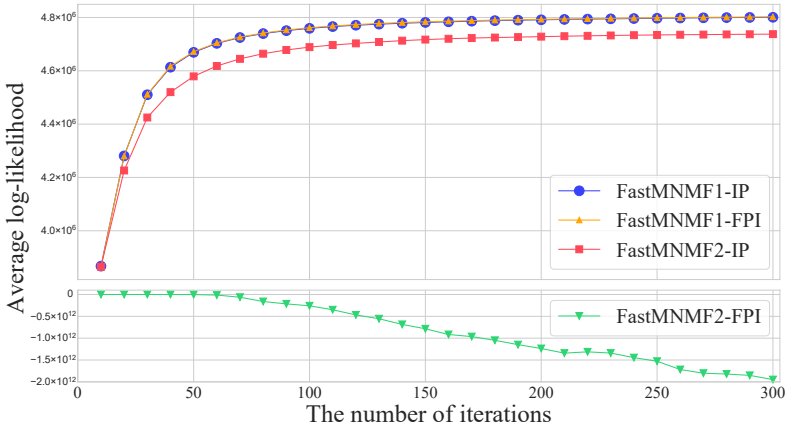


Figure 4.11: The evolutions of the average log-likelihoods obtained by FastMNMF1, FastMNMF2, FastMNMF1-FPI, and FastMNMF2-FPI with the circular initialization.

with a higher degree of freedom in the subsequent step.

Comparing FastMNMF with FastMNMF-FPI, FastMNMF1-FPI (7.9 dB) performed slightly better than FastMNMF1 (7.4 dB), while FastMNMF2-FPI failed in all cases (1.1 dB). Fig. 4.11 shows the evolutions of the average log-likelihoods obtained by FastMNMF and FastMNMF-FPI with the circular initialization. The log-likelihoods of FastMNMF1 were almost the same as those of FastMNMF1-FPI and higher than those of FastMNMF2 because FastMNMF1 has larger model complexity than FastMNMF2. While the log-likelihoods of FastMNMF1, FastMNMF1-

FPI, and FastMNMF2 increased monotonically, those of FastMNMF2-FPI tended to decrease because the FPI method has no guarantee to increase the likelihood.

4.7.7 RC-FastMNMF for Speech Enhancement

We evaluated the effectiveness of rank-constrained FastMNMF in speech enhancement.

Experimental Conditions

We used the evaluation dataset of CHiME3 [135], which contains 1320 noisy speech signals simulated for a tablet with six microphones in four types of noisy environments: bus (BUS), cafe (CAF), pedestrian area (PED), and street junction (STR). 25 utterances were selected randomly for each environment (100 utterances in total). The average SDR of the input noisy signals (the fifth channel) was 7.5 dB. $M = 5$ channels excluding the second channel behind the tablet were used and $N = 5$ sources (one speech source and four noise sources) were assumed to exist. In this experiment, STFT with a shifting interval of 256 points and a Hann window of 1024 points was used ($F = 513$).

We evaluated the rank-constrained FastMNMF, named RC-FastMNMF$_{(1,M)}$ (Section 4.6.2), where the SCMs of source 1 (directional speech) were restricted to rank-1 matrices and those of source $n \in \{2, \cdots, N\}$ (diffuse noise) were left as full-rank matrices. For comparison, we tested RC-FastMNMF$_{(1,M-1)}$ with the rank-1 SCMs of source 1 and the rank-$(M-1)$ SCMs of source $n \in \{2, \cdots, N\}$ obtained by initializing $\tilde{\mathbf{g}}_{n(\geq 2)f}$ with $[0, 1, \ldots, 1]$. We also tested a speech enhancement method based on ILRMA [79]. Specifically, the rank-1 SCMs of speech and the rank-$(M-1)$ SCMs of noise were estimated with ILRMA and the missing rank-1 SCMs of noise and the PSDs of speech and noise were then estimated in an independent step. In addition, we tested two-step RC-FastMNMF, where $\mathbf{Q}$ was estimated with ILRMA and $\mathbf{W}$, $\mathbf{H}$, and $\tilde{\mathbf{G}}$ were then estimated (Section 4.2.4). Note that [79] is similar to two-step RC-FastMNMF1. A main difference is that in [79] an inverse gamma prior distribution with a shape of 0.7 and a scale of 10^{-16} was put on the speech PSDs. These methods have the same advantage that

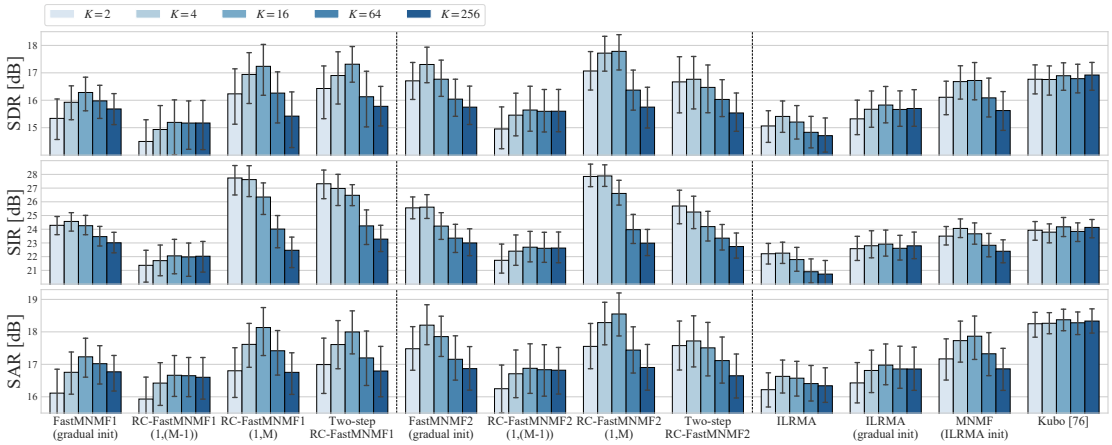


Figure 4.12: SDRs, SIRs, and SARs obtained by RC-FastMNMF, FastMNMF, ILRMA, and MNMF in speech enhancement.

the rank of the SCMs of each source can be specified explicitly according to its directivity.

As general-purpose BSS methods, we tested vanilla FastMNMF with the gradual initialization method, MNMF with ILRMA-based initialization, and ILRMA with the diagonal or gradual initialization method. For evaluation, the most dominant source in terms of the average power was selected as target speech from N estimated sources.

Experimental Results

Fig. 4.12 shows the SDRs, SIRs, and SARs of the compared methods averaged over the 100 utterances. In almost all versions of rank-constrained FastMNMF, the SDRs and SARs were maximized when $K = 16$, and the SIRs were maximized when $K = 2$. RC-FastMNMF2$_{(1,M)}$ achieved the highest SDR (17.8 dB) and outperformed RC-FastMNMF2$_{(1,M-1)}$ (15.6 dB) and ILRMA with $K = 16$ and the gradual initialization method (15.8 dB). Similarly, RC-FastMNMF1$_{(1,M)}$ (17.2 dB) outperformed RC-FastMNMF1$_{(1,M-1)}$ (15.2 dB). This indicates that the full-rankness of the noise SCMs was important for speech enhancement. RC-FastMNMF2$_{(1,M)}$ outperformed vanilla FastMNMF2 with $K = 4$ and the gradual initialization method (17.3 dB). When $\tilde{\mathbf{g}}_n$ of each source n was initialized to a one-hot-like vector, the noise SCMs estimated by FastMNMF2 were often

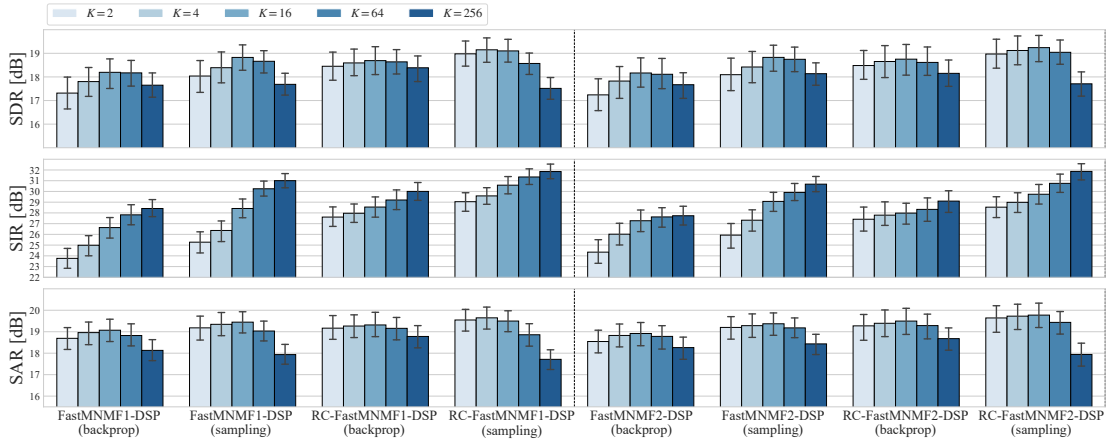


Figure 4.13: SDRs, SIRs, and SARs obtained by RC-FastMNMF-DSP and FastMNMF-DSP in speech enhancement.

close to rank-deficient matrices. RC-FastMNMF2$_{(1,M)}$ outperformed two-step RC-FastMNMF1 with $K = 16$ (17.3 dB) and two-step RC-FastMNMF2 with $K = 4$ (16.8 dB). This indicates the importance of jointly estimating $\mathbf{Q}$, $\mathbf{W}$, $\mathbf{H}$, and $\tilde{\mathbf{G}}$, as reported in Section 4.7.6.

We also evaluated FastMNMF-DSP (Section 4.4) with sampling and backpropagation algorithms for estimating latent variables and their rank-constrained variants called RC-FastMNMF-DSP. The deep speech generative model was trained in the same way as the experiment in Section 3.5. Fig. 4.13 shows the SDRs, SIRs, and SARs of these methods. RC-FastMNMF2-DSP with $K = 16$ and sampling algorithm achieved the highest SDR (19.2 dB), and RC-FastMNMF1-DSP with $K = 4$ or $K = 16$ and sampling algorithm achieved almost the same performance (19.1 dB). Because of the powerful speech model, even the methods based on FastMNMF1 was able to avoid bad local optima.

4.7.8 RC-FastMNMF for Speech Separation

We evaluated the effectiveness of rank-constrained FastMNMF in speech separation using a dataset recorded in a real environment.

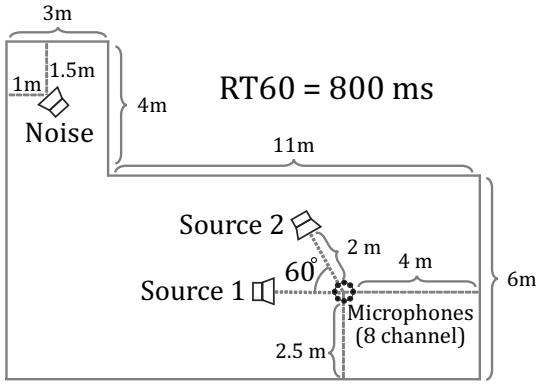


Figure 4.14: Recording condition of the real data in Section 4.7.8.

Experimental Conditions

An eight-channel microphone array ($M = 8$) and three loudspeakers corresponding to two speech sources and one noise source were put in a spacious, heavily-echoic room with $\mathrm{RT}_{60} = 800$ ms (Fig. 4.14). The loudspeaker placed far away from the microphones was used for emitting a noise signal to the wall to simulate diffuse noise. We randomly selected 20 clean speech signals from the WSJ-0 corpus and four noise signals from the CHiME3 evaluation dataset. To obtain ground-truth images, these signals were recorded individually and 20 mixtures were synthesized by superimposing randomly-selected speech and noise signals, where the signal-to-noise ratio (SNR) was set to 0 dB. The average SDR of the input mixture signals (the first channel) was -4.1 dB.

We tested RC-FastMNMF, FastMNMF with the gradual initialization method, and two-step RC-FastMNMF, where $N = 8$ sources were assumed to exist in order to deal with heavy reverberation (a number of virtual sources were considered). In RC-FastMNMF and two-step RC-FastMNMF, the SCMs of two speech sources were restricted to rank-1 matrices and those of six noise sources were full-rank matrices. For comparison, we tested ILRMA with the diagonal or gradual initialization method, MNMF with ILRMA-based initialization, and ILRMA with the clustering mechanism [78]. For evaluation, two dominant sources in terms of the average power were selected from N sources as target speech sources.

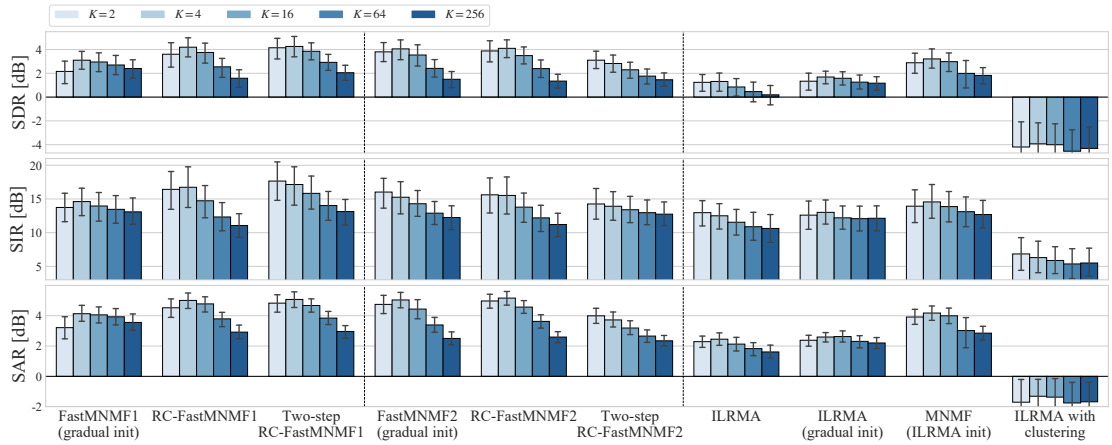


Figure 4.15: SDRs, SIRs, and SARs obtained by RC-FastMNMF, FastMNMF, ILRMA, and MNMF in speech separation.

Experimental Results

Fig. 4.15 shows the SDRs, SIRs, and SARs averaged over the 20 mixtures. In FastMNMF and RC-FastMNMF, the SDRs were maximized when $K = 4$. In this experiment, we found no significant difference between RC-FastMNMF2 and FastMNMF2 (4.1 dB) because the rank-1 assumption on the SCMs of speech was violated by the heavy reverberation, which was much longer than the STFT window size. In contrast, RC-FastMNMF1 (4.2 dB) outperformed FastMNMF1 (3.1 dB) because the inter-frequency weight sharing in two speech sources ($\tilde{\mathbf{g}}_{1f} = \mathbf{e}_n$ and $\tilde{\mathbf{g}}_{2f} = \mathbf{e}_2$ for any f) helped parameter estimation as in FastMNMF2. Two-step RC-FastMNMF1 with $K = 4$ (4.3 dB) outperformed RC-FastMNMF2 (4.1 dB), although $\mathbf{Q}$ estimated by ILRMA was considered to be sub-optimal, as discussed in Sections 4.7.6 and 4.7.7. This indicates that RC-FastMNMF1 with $\tilde{\mathbf{g}}_{nf}$ might be more suitable than RC-FastMNMF2 with $\tilde{\mathbf{g}}_n$ for representing strong diffuse noise in a highly reverberant environment.

4.8 Summary

In this chapter, we proposed a versatile and computationally-efficient BSS method called FastMNMF based on directivity-aware jointly-diagonalizable full-rank SCMs. FastMNMF is a special case of MNMF based on unconstrained full-rank

SCMs [27]. More specifically, at each frequency bin, we represent the full-rank SCMs of sources as the weighted sums of common rank-1 matrices corresponding to different directions, resulting in FastMNMF1. Given that the directional feature of each source should be consistent over frequency bins, we make the weights of FastMNMF1 shared over frequency bins, resulting in FastMNMF2. To avoid bad local optima in iterative parameter optimization, we proposed and experimentally compared four initialization methods. To explicitly consider the directivity or diffuseness of each source, we further derived rank-constrained FastMNMF that enables us to individually specify the ranks of SCMs.

In a speech separation experiment, we confirmed that FastMNMF2 outperformed FastMNMF1, especially for larger numbers of microphones and frequency bins. We found that the circular and gradual initialization methods worked well. In a speech enhancement experiment, RC-FastMNMF2 with rank-1 speech SCMs and full-rank noise SCMs achieved the best performance.

Chapter 5

Joint Multichannel Speech Separation and Dereverberation Based on an ARMA Model

5.1 Introduction

In Chapters 3 and 4, we proposed extensions of MNMF using a DNN-based speech model and a jointly-diagonalizable (JD) full-rank spatial model. The goal of these methods is to extract source images $\mathbf{x}_{nft}$, and reverberations are included in the extracted source images. Since the reverberation is known to be harmful for speech intelligibility and ASR performance [115, 116], in this chapter, we tackle joint source separation and dereverberation that outputs direct signal of each source given reverberant mixtures.

We propose a joint blind source separation and dereverberation method called ARMA-FastMNMF2 that extends FastMNMF2 by explicitly modeling reverberations. Reverberations are often represented by a moving average (MA) model [15, 17]. When the reverberation time is long, however, the number of parameters becomes quite large. To alleviate this problem, an autoregressive (AR) model [18, 19] is introduced to represent the late part of the reverberation (late reverberation), and the MA model is used to mainly represent the early part (early reflection), resulting in the ARMA model [46]. The AR model can represent infinitely-long reverberations with a finite tap length in theory, and has been used in one of the most successful dereverberation methods called weighted

prediction error (WPE) [47]. If the AR model is used for representing the early reflections, however, it may also represent the direct speech signals because of the correlations inherent in the speech signals. Since ARMA-FastMNMF2 is based on the JD full-rank spatial model, it can be used even in an overdetermined case and deal with diffuse noise unlike independent low-rank matrix analysis (ILRMA) [28] and its extension called AR-ILRMA [118, 148], which integrates the AR model with ILRMA.

To derive an efficient update rules for the AR coefficients, we introduce the joint-diagonalization constraint on the MA model. Although the joint source separation and dereverberation methods based on the ARMA model and full-rank spatial model have already been proposed in [46, 120], the computational cost for estimating the AR coefficients is quite expensive due to the unconstrained SCMs, especially when the tap length of the AR model is long. The joint-diagonalization constraint makes it possible to jointly estimate the AR coefficients and diagonalizers in a computationally highly efficient manner as in [148]. Because the MA model may also represent a part of the direct speech signals, we further introduce the rank-constraint to the SCMs of the MA model to keep them away from those of the direct signals. Moreover, we derive ARMA-FastMNMF2-DSP by integrating a DNN-based speech model into ARMA-FastMNMF2. In our experiments, we confirmed the effectiveness of ARMA-FastMNMF2 compared to MA-, and AR-FastMNMF2 in speech separation tasks, and that of ARMA-FastMNMF2-DSP in speech enhancement tasks.

5.2 ARMA Model for Reverberation

When the reverberation is longer than the window size of short-time Fourier transform (STFT), we assume that the image of source n, $\mathbf{x}_{nft}$, is written using a moving average (MA) model as follows:

$$\mathbf{x}_{nft} = \sum_{l=0}^{L-1} \mathbf{a}_{nfl} s_{n,f,t-l} = \mathbf{a}_{nf0} s_{nft} + \sum_{l=1}^{\Delta-1} \mathbf{a}_{nfl} s_{n,f,t-l} + \sum_{l=\Delta}^{L-1} \mathbf{a}_{nfl} s_{n,f,t-l} \tag{5.1}$$

$$\triangleq \mathbf{d}_{nft} + \mathbf{r}_{nft}^{\text{early}} + \mathbf{r}_{nft}^{\text{late}}, \tag{5.2}$$

where L is the length of the impulse response and Δ indicates the boundary between early reflections and late reverberations and is called *delay* as in [19]. $\mathbf{a}_{nfl}$ is the STFT coefficients of the impulse response of source n at frequency f and time l. $\mathbf{d}_{nft}$, $\mathbf{r}_{nft}^{\text{early}}$, and $\mathbf{r}_{nft}^{\text{late}}$ correspond to the direct signal, early reflections, and late reverberations of source n, respectively. When the reverberation time is long, the parameter L becomes large, resulting in a large number of parameters. To alleviate this problem, we represent the late reverberations with an autoregressive (AR) model. As in [47], r_{nftm}^{late}, the m-th element of $\mathbf{r}_{nft}^{\text{late}}$, is rewritten using a_{nflm}, the m-th element of $\mathbf{a}_{nfl}$, as follows:

$$
\begin{aligned}
r_{nftm}^{\text{late}} &= \sum_{l=\Delta}^{L-1} a_{nflm} s_{n,f,t-l} = \hat{\mathbf{a}}_{nfm}^{\mathsf{T}} \hat{\mathbf{s}}_{n,f,t-\Delta} && (5.3)\\
&= \hat{\mathbf{a}}_{nfm}^{\mathsf{T}} (\hat{\mathbf{A}}_{nf}^{\mathsf{T}} \hat{\mathbf{A}}_{nf})^{-1} \hat{\mathbf{A}}_{nf}^{\mathsf{T}} \hat{\mathbf{A}}_{nf} \hat{\mathbf{s}}_{n,f,t-\Delta} && (5.4)\\
&= \hat{\mathbf{a}}_{nfm}^{\mathsf{T}} (\hat{\mathbf{A}}_{nf}^{\mathsf{T}} \hat{\mathbf{A}}_{nf})^{-1} \hat{\mathbf{A}}_{nf}^{\mathsf{T}} \hat{\mathbf{x}}_{n,f,t-\Delta} && (5.5)\\
&= \hat{\mathbf{b}}_{nfm}^{\mathsf{T}} \hat{\mathbf{x}}_{n,f,t-\Delta} = \sum_{l=\Delta}^{\Delta+L_{\text{AR}}-1} \hat{\mathbf{b}}_{nflm}^{\mathsf{T}} \mathbf{x}_{n,f,t-l}, && (5.6)
\end{aligned}
$$

where $\hat{\mathbf{a}}_{nfm} \triangleq [a_{nf\Delta m}, \ldots, a_{n,f,L-1,m}, \mathbf{0}_{L_{\text{AR}}+\Delta-1}^{\mathsf{T}}]^{\mathsf{T}} \in \mathbb{C}^{L'}$, $\hat{\mathbf{s}}_{nft} \triangleq [s_{nft}, \ldots, s_{n,f,t-L'+1}]^{\mathsf{T}} \in \mathbb{C}^{L'}$, $L' = L + L_{\text{AR}} - 1$, and L_{AR} is the tap length of the AR model. $\hat{\mathbf{b}}_{nfm}$, $\hat{\mathbf{b}}_{nflm}$, $\hat{\mathbf{x}}_{nft}$ and $\hat{\mathbf{A}}_{nf}$ are given by

$$
\begin{aligned}
\hat{\mathbf{b}}_{nfm} &\triangleq \hat{\mathbf{a}}_{nfm}^{\mathsf{T}} (\hat{\mathbf{A}}_{nf}^{\mathsf{T}} \hat{\mathbf{A}}_{nf})^{-1} \hat{\mathbf{A}}_{nf}^{\mathsf{T}} && (5.7)\\
&\triangleq [b_{nfm1\Delta}, \ldots, b_{n,f,m,1,\Delta+L_{\text{AR}}-1}, \ldots, b_{nfmM\Delta}, \ldots, b_{n.f,m,M,\Delta+L_{\text{AR}}-1}]^{\mathsf{T}}\\
&\in \mathbb{C}^{ML_{\text{AR}}}, && (5.8)\\
\hat{\mathbf{b}}_{nflm} &\triangleq [b_{nfm1l}, \ldots, b_{nfmMl}]^{\mathsf{T}} \in \mathbb{C}^{M}, && (5.9)\\
\hat{\mathbf{x}}_{nft} &\triangleq [x_{nft1}, \ldots, x_{n,f,t-L_{\text{AR}}+1,1}, \ldots, x_{nftM}, \ldots, x_{n,f,t-L_{\text{AR}}+1,M}]^{\mathsf{T}} \in \mathbb{C}^{ML_{\text{AR}}}, && (5.10)\\
\hat{\mathbf{A}}_{nf} &\triangleq [\hat{\mathbf{A}}_{nf1}^{\mathsf{T}}, \ldots, \hat{\mathbf{A}}_{nfM}^{\mathsf{T}}]^{\mathsf{T}} \in \mathbb{C}^{ML_{\text{AR}} \times L'}, && (5.11)\\
\hat{\mathbf{A}}_{nfm} &\triangleq \begin{pmatrix}
a_{nf0m} & \cdots & a_{n,f,L-1,m} & 0 & \cdots & 0\\
0 & a_{nf0m} & \cdots & a_{n,f,L-1,m} & 0 & \cdots & 0\\
\vdots & & \ddots & & & & \vdots\\
0 & & \cdots & 0 & a_{nf0m} & \cdots & a_{n,f,L-1,m}
\end{pmatrix}\\
&\in \mathbb{C}^{L_{\text{AR}} \times L'}, && (5.12)
\end{aligned}
$$

where $\hat{\mathbf{A}}_{nf} \in \mathbb{C}^{ML_{\mathrm{AR}} \times L'}$ has to be a full column rank matrix ($ML_{\mathrm{AR}} \geq L' = L + L_{\mathrm{AR}} - 1$) to make $\hat{\mathbf{A}}_{nf}^{\mathsf{T}} \hat{\mathbf{A}}_{nf}$ a non-singular matrix, *i.e.*, $L_{\mathrm{AR}} \geq (L-1)/(M-1)$.

When there are multiple sound sources, we assume $\mathbf{x}_{ft}$ is written using the MA model as follows:

$$\mathbf{x}_{ft} = \sum_{n=1}^{N} \left(\mathbf{a}_{nf0} s_{nft} + \sum_{l=1}^{\Delta-1} \mathbf{a}_{nfl} s_{n,f,t-l} \right) + \sum_{n=1}^{N} \sum_{l=\Delta}^{L-1} \mathbf{a}_{nfl} s_{n,f,t-l} \tag{5.13}$$

$$\triangleq \sum_{n=1}^{N} (\mathbf{d}_{nft} + \mathbf{r}_{nft}^{\text{early}}) + \mathbf{r}_{ft}^{\text{late}}, \tag{5.14}$$

where $\mathbf{r}_{ft}^{\text{late}}$ is the late reverberation of all the sources. As in Eq. (5.6), r_{ftm}^{late} is rewritten as follows:

$$r_{ftm}^{\text{late}} = \sum_{n=1}^{N} \sum_{l=\Delta}^{L-1} a_{nflm} s_{n,f,t-l} = \hat{\mathbf{a}}_{fm}^{\mathsf{T}} \hat{\mathbf{s}}_{f,t-\Delta} \tag{5.15}$$

$$= \hat{\mathbf{a}}_{fm}^{\mathsf{T}} (\hat{\mathbf{A}}_f^{\mathsf{T}} \hat{\mathbf{A}}_f)^{-1} \hat{\mathbf{A}}_f^{\mathsf{T}} \hat{\mathbf{A}}_f \hat{\mathbf{s}}_{f,t-\Delta} \tag{5.16}$$

$$= \hat{\mathbf{a}}_{fm}^{\mathsf{T}} (\hat{\mathbf{A}}_f^{\mathsf{T}} \hat{\mathbf{A}}_f)^{-1} \hat{\mathbf{A}}_f^{\mathsf{T}} \hat{\mathbf{x}}_{f,t-\Delta} \tag{5.17}$$

$$= \hat{\mathbf{b}}_{fm}^{\mathsf{T}} \hat{\mathbf{x}}_{f,t-\Delta} = \sum_{l=\Delta}^{\Delta+L_{\mathrm{AR}}-1} \hat{\mathbf{b}}_{flm}^{\mathsf{T}} \mathbf{x}_{f,t-l} \tag{5.18}$$

where $\hat{\mathbf{s}}_{ft} \triangleq [\hat{\mathbf{s}}_{1ft}^{\mathsf{T}}, \ldots, \hat{\mathbf{s}}_{Nft}^{\mathsf{T}}]^{\mathsf{T}} \in \mathbb{C}^{NL'}$ and $\hat{\mathbf{a}}_{fm} \triangleq [\hat{\mathbf{a}}_{1fm}^{\mathsf{T}}, \ldots, \hat{\mathbf{a}}_{Nfm}^{\mathsf{T}}]^{\mathsf{T}} \in \mathbb{C}^{NL'}$. $\hat{\mathbf{c}}_{fm}$, $\hat{\mathbf{c}}_{flm}$, $\hat{\mathbf{A}}_f$, and $\hat{\mathbf{x}}_{ft}$ are given by

$$\hat{\mathbf{b}}_{fm} \triangleq \hat{\mathbf{a}}_{fm}^{\mathsf{T}} (\hat{\mathbf{A}}_f^{\mathsf{T}} \hat{\mathbf{A}}_f)^{-1} \hat{\mathbf{A}}_f^{\mathsf{T}} \tag{5.19}$$

$$\triangleq [b_{fm1\Delta}, \ldots, b_{f,m,1,\Delta+L_{\mathrm{AR}}-1}, \ldots, b_{fmM\Delta}, \ldots, b_{f,m,M,\Delta+L_{\mathrm{AR}}-1}]^{\mathsf{T}} \in \mathbb{C}^{ML_{\mathrm{AR}}}, \tag{5.20}$$

$$\hat{\mathbf{b}}_{flm} \triangleq [b_{fm1l}, \ldots, b_{fmMl}]^{\mathsf{T}} \in \mathbb{C}^{M}, \tag{5.21}$$

$$\hat{\mathbf{A}}_f \triangleq [\hat{\mathbf{A}}_{1f}, \ldots, \hat{\mathbf{A}}_{Nf}] \in \mathbb{C}^{ML_{\mathrm{AR}} \times NL'}, \tag{5.22}$$

$$\hat{\mathbf{x}}_{ft} = [x_{ft1}, \ldots, x_{f,t-L_{\mathrm{AR}}+1,1}, \ldots, x_{ftM}, \ldots, x_{f,t-L_{\mathrm{AR}}+1,M}] \in \mathbb{C}^{ML_{\mathrm{AR}}}, \tag{5.23}$$

where $\hat{\mathbf{A}}_f \in \mathbb{C}^{ML_{\mathrm{AR}} \times NL'}$ has to be a full column rank matrix to make $\hat{\mathbf{A}}_f^{\mathsf{T}} \hat{\mathbf{A}}_f$ a non-singular matrix, that is, $L_{\mathrm{AR}} \geq N(L-1)/(M-N)$. Fig. 5.1 shows the relationship between $\hat{\mathbf{s}}_{f,t-\Delta}$ and $\hat{\mathbf{x}}_{f,t-\Delta}$.

Using Eq. (5.18) and $\hat{\mathbf{B}}_{fl} \triangleq [\hat{\mathbf{b}}_{fl1}, \ldots, \hat{\mathbf{b}}_{flM}]^{\mathsf{T}} \in \mathbb{C}^{M \times M}$, Eq. (5.14) is rewritten

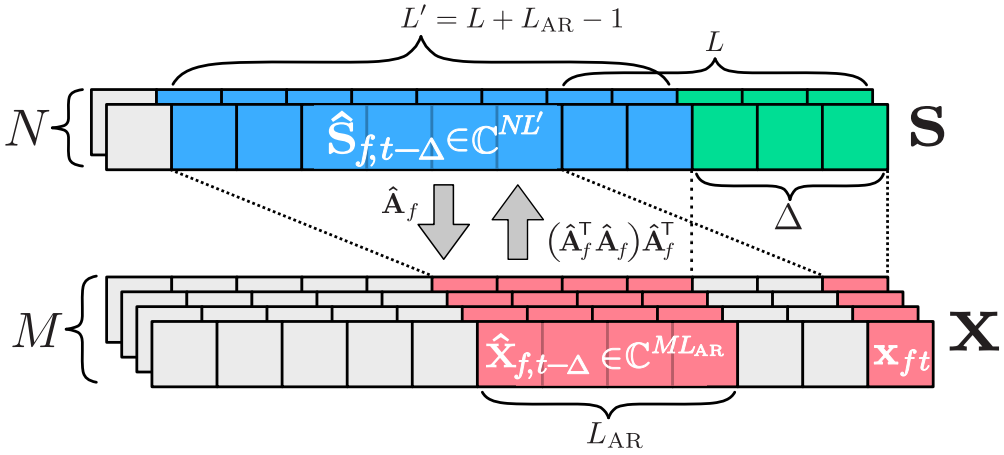


Figure 5.1: The relationship of $\hat{\mathbf{s}}_{f,t-\Delta}$ and $\hat{\mathbf{x}}_{f,t-\Delta}$.

as follows:

$$\mathbf{x}_{ft} = \sum_{n=1}^{N} \left(\mathbf{a}_{nf0} s_{nft} + \sum_{l=1}^{\Delta-1} \mathbf{a}_{nfl} s_{n,f,t-l} \right) + \sum_{l=\Delta}^{\Delta+L_{\mathrm{AR}}-1} \hat{\mathbf{B}}_{fl} \mathbf{x}_{f,t-l}. \tag{5.24}$$

Since the early reflection is represented with the MA model and the late reverberation is represented with the AR model, Eq. (5.24) is called an ARMA model. When $N/(M-N) \leq 1$, according to Eq. (5.18), $\mathbf{x}_{ft}$ represented by the MA model with a certain L can be represented by the ARMA model with $L_{\mathrm{AR}} < L$. Thus, although the AR model apparently represent the late reverberation of all the sources as a whole, it can represent the *source-wise* reverberations as the MA model. When $\hat{\mathbf{B}}_{fl}$ is not limited to one calculated with Eq. (5.19), the ARMA model does not always correspond to an MA model, and it represents long reverberations even if L_{AR} is small. If $\Delta = 1$ is used, all the reverberations can be represented by the AR model in theory. This, however, causes over-whitening of the direct signals. Thus, $\Delta = 2$ or 3 is used in practice, and the MA model represents the remaining reverberations.

5.3 FastMNMF2 with an ARMA Model (ARMA-FastMNMF2)

This section explains the joint separation and dereverberation method based on the weight-shared jointly-diagonalizable (WJD) spatial model and auto regressive

moving average (ARMA) model.

5.3.1 Formulation

We formulate the observed reverberant mixture $\mathbf{x}_{ft}$ using the ARMA-based reverberation model as follows:

$$\mathbf{x}_{ft} = \sum_{n=1}^{N} \mathbf{a}_{nf0} s_{nft} + \sum_{n=1}^{N} \sum_{l=1}^{L_{\mathrm{MA}}} \mathbf{a}_{nfl} s_{n,f,t-l} + \sum_{l'=\Delta}^{\Delta+L_{\mathrm{AR}}-1} \mathbf{B}_{fl'} \mathbf{x}_{f,t-l'}, \tag{5.25}$$

$$= \sum_{n=1}^{N} \sum_{l \in \mathbb{L}_{\mathrm{MA}}} \mathbf{a}_{nfl} s_{n,f,t-l} + \sum_{l' \in \mathbb{L}_{\mathrm{AR}}} \mathbf{B}_{fl'} \mathbf{x}_{f,t-l'}, \tag{5.26}$$

where $L_{\mathrm{MA}} (\geq 1)$ is the tap length for the MA model, $\mathbb{L}_{\mathrm{MA}} \triangleq \{0, 1, \ldots, L_{\mathrm{MA}}\}$, and $\mathbb{L}_{\mathrm{AR}} \triangleq \{\Delta, \ldots, \Delta + L_{\mathrm{AR}} - 1\}$. Although in Eq. (5.24) the MA model represents only the early reflections and the AR model represents only the late reverberations, here, the MA model is also used to represent the residual late reverberations that cannot be represented by the AR model by setting $L_{\mathrm{MA}} \geq \Delta$.

From Eq. (1.2) and the reproductive property of the Gaussian distribution, we have

$$\sum_{n=1}^{N} \sum_{l \in \mathbb{L}_{\mathrm{MA}}} \mathbf{a}_{nfl} s_{n,f,t-l} \sim \mathcal{N}_{\mathbb{C}} \left(\mathbf{0}, \sum_{n=1}^{N} \sum_{l \in \mathbb{L}_{\mathrm{MA}}} \lambda_{n,f,t-l} \mathbf{G}_{nfl} \right), \tag{5.27}$$

where $\mathbf{G}_{nfl} = \mathbf{a}_{nfl} \mathbf{a}_{nfl}^{\mathsf{H}}$. Here, we assume that $\{\mathbf{G}_{nfl}\}_{n,l}$ are WJD full-rank matrices to derive the update rule as follows:

$$\forall n \in \{1, \ldots, N\}, \quad \forall l \in \mathbb{L}_{\mathrm{MA}}, \quad \mathbf{Q}_f \mathbf{G}_{nfl} \mathbf{Q}_f^{\mathsf{H}} = \mathrm{Diag}\left(\tilde{\mathbf{g}}_{nl}\right), \tag{5.28}$$

where $\tilde{\mathbf{g}}_{nl} = [\tilde{g}_{nl1}, \ldots, \tilde{g}_{nlM}] \in \mathbb{R}_{+}^{M}$ is shared over all frequency bins as in FastMNMF2. Thus, we call this method *ARMA-FastMNMF2*. The joint diagonalization constraint indicates that $N(L_{\mathrm{MA}} + 1)$ matrices $\{\mathbf{G}_{nfl}\}_{n,l}$ are represented by the weighted sum of M rank-1 matrices $\{\mathbf{u}_{fm} \mathbf{u}_{fm}^{\mathsf{H}}\}_{m=1}^{M}$, where $\mathbf{u}_{fm}$ is the m-th column vector of $\mathbf{U}_f = \mathbf{Q}_f^{-1}$. From Eqs. (5.26), (5.27), and (5.28), we have

$$\mathbf{x}_{ft} \,|\, \{\mathbf{x}_{f,t-l}\}_{l \in \mathbb{L}_{\mathrm{AR}}} \sim \mathcal{N}_{\mathbb{C}} \left(\sum_{l' \in \mathbb{L}_{\mathrm{AR}}} \mathbf{B}_{fl'} \mathbf{x}_{f,t-l'}, \mathbf{Q}_f^{-1} \left(\sum_{n=1}^{N} \sum_{l \in \mathbb{L}_{\mathrm{MA}}} \lambda_{n,f,t-l} \mathrm{Diag}(\tilde{\mathbf{g}}_{nl}) \right) \mathbf{Q}_f^{-\mathsf{H}} \right) \tag{5.29}$$

$$\triangleq \mathcal{N}_{\mathbb{C}}\left(\sum_{l'\in\mathbb{L}_{\mathrm{AR}}}\mathbf{B}_{fl'}\mathbf{x}_{f,t-l'}, \mathbf{Q}_f^{-1}\mathrm{Diag}(\tilde{\mathbf{y}}_{ft})\,\mathbf{Q}_f^{-\mathsf{H}}\right) \tag{5.30}$$

$$\triangleq \mathcal{N}_{\mathbb{C}}\left(\sum_{l'\in\mathbb{L}_{\mathrm{AR}}}\mathbf{B}_{fl'}\mathbf{x}_{f,t-l'}, \mathbf{Y}_{ft}\right), \tag{5.31}$$

where $\tilde{\mathbf{y}}_{ft} \triangleq [\tilde{y}_{ft1},\ldots,\tilde{y}_{ftM}]^{\mathsf{T}}$ and $\tilde{y}_{ftm} \triangleq \sum_{n=1}^{N}\sum_{l\in\mathbb{L}_{\mathrm{MA}}}\lambda_{n,f,t-l}\tilde{g}_{nlm}$.

To avoid the scale ambiguity, we put normalization constraints on $\mathbf{Q}$, $\tilde{\mathbf{G}}$, and $\mathbf{W}$ as follows:

$$\mathrm{tr}(\mathbf{Q}_f\mathbf{Q}_f^{\mathsf{H}}) = M, \tag{5.32}$$

$$\sum_{l\in\mathbb{L}_{\mathrm{MA}}}\sum_{m=1}^{M}\tilde{g}_{nlm} = 1, \tag{5.33}$$

$$\sum_{f=1}^{F} w_{nkf} = 1. \tag{5.34}$$

5.3.2 Optimization

The parameters $\boldsymbol{\Theta} \triangleq \{\mathbf{W},\mathbf{H},\mathbf{Q},\tilde{\mathbf{G}},\mathbf{B}\}$ are estimated such that the log-likelihood $\log p(\mathbf{X}|\boldsymbol{\Theta})$ is maximized. From Eq. (5.29), it is given by

$$\log p\left(\mathbf{X}|\boldsymbol{\Theta}\right) = \sum_{f,t=1}^{F,T}\log p\left(\mathbf{x}_{ft}|\boldsymbol{\Theta},\{\mathbf{x}_{f,t-l'}\}_{l'\in\mathbb{L}_{\mathrm{AR}}}\right) \tag{5.35}$$

$$\propto -\sum_{f,t,m=1}^{F,T,M}\frac{|\mathbf{q}_{fm}^{\mathsf{H}}(\mathbf{x}_{ft}-\sum_{l'\in\mathbb{L}_{\mathrm{AR}}}\mathbf{B}_{fl'}\mathbf{x}_{f,t-l'})|^2}{\tilde{y}_{ftm}}$$

$$-\sum_{f,t,m=1}^{F,T,M}\log\tilde{y}_{ftm} + T\sum_{f=1}^{F}\log|\mathbf{Q}_f\mathbf{Q}_f^{\mathsf{H}}|, \tag{5.36}$$

$$= -\sum_{f,t,m=1}^{F,T,M}\left(\frac{|\mathbf{q}_{fm}^{\mathsf{H}}\bar{\mathbf{d}}_{ft}|^2}{\tilde{y}_{ftm}} + \log\tilde{y}_{ftm}\right) + T\sum_{f=1}^{F}\log|\mathbf{Q}_f\mathbf{Q}_f^{\mathsf{H}}| \tag{5.37}$$

$$= -\sum_{f,t,m=1}^{F,T,M}\left(\frac{\tilde{d}_{ftm}}{\tilde{y}_{ftm}} + \log\tilde{y}_{ftm}\right) + T\sum_{f=1}^{F}\log|\mathbf{Q}_f\mathbf{Q}_f^{\mathsf{H}}|, \tag{5.38}$$

where $\bar{\mathbf{d}}_{ft} \triangleq \mathbf{x}_{ft} - \sum_{l'\in\mathbb{L}_{\mathrm{AR}}}\mathbf{B}_{fl'}\mathbf{x}_{f,t-l'}$ and $\tilde{d}_{ftm} \triangleq |\mathbf{q}_{fm}^{\mathsf{H}}\bar{\mathbf{d}}_{ft}|^2$.

On condition that $\mathbf{B}$ is given, since Eq. (5.38) has the same form as the log-likelihood of FastMNMF2 given by Eq. (4.22), $\mathbf{Q}$, $\mathbf{W}$, $\mathbf{H}$, and $\tilde{\mathbf{G}}$ can be updated

by almost the same way as FastMNMF2. $\mathbf{Q}_f$ is updated using the IP method as

$$\mathbf{V}_{fm} \triangleq \frac{1}{T}\sum_{t=1}^{T}\frac{\bar{\mathbf{d}}_{ft}\bar{\mathbf{d}}_{ft}^{\mathsf{H}}}{\tilde{y}_{ftm}}, \tag{5.39}$$

$$\mathbf{q}_{fm} \leftarrow (\mathbf{Q}_f\mathbf{V}_{fm})^{-1}\mathbf{e}_m, \tag{5.40}$$

$$\mathbf{q}_{fm} \leftarrow (\mathbf{q}_{fm}^{\mathsf{H}}\mathbf{V}_{fm}\mathbf{q}_{fm})^{-\frac{1}{2}}\mathbf{q}_{fm}, \tag{5.41}$$

where $\mathbf{e}_m$ is a one-hot vector whose m-th element is 1. We use a minorization-maximization (MM) algorithm to estimate $\mathbf{W}$, $\mathbf{H}$, and $\tilde{\mathbf{G}}$. $\mathbf{W}$, $\mathbf{H}$, and $\tilde{\mathbf{G}}$ depend only on the first two terms of Eq. (5.38), and the lower bound is derived using a Jensen's inequality and first-order Taylor expansion as follows:

$$-\frac{\tilde{d}_{ftm}}{\tilde{y}_{ftm}} - \log\tilde{y}_{ftm} \tag{5.42}$$

$$\geq -\sum_{n,k=1}^{N,K}\sum_{l\in\mathbb{L}_{\mathrm{MA}}}\left(\frac{\alpha_{ftmnlk}^2\tilde{d}_{ftm}}{w_{nkf}h_{n,k,t-l}\tilde{g}_{nlm}}\right) - \frac{\tilde{y}_{ftm}}{\beta_{ftm}} - \log\beta_{ftm} + 1 \tag{5.43}$$

$$\triangleq \mathcal{L}, \tag{5.44}$$

where the equality holds if and only if $\alpha_{ftmnlk} = w_{nkf}h_{n,k,t-l}\tilde{g}_{nlm}\tilde{y}_{ftm}^{-1}$ and $\beta_{ftm} = \tilde{y}_{ftm}$. Letting the partial derivatives of $\mathcal{L}$ with respect to $\mathbf{W}$, $\mathbf{H}$, and $\tilde{\mathbf{G}}$ equal to zero, the closed-form multiplicative update (MU) rules are obtained as follows:

$$w_{nkf} \leftarrow w_{nkf}\sqrt{\frac{\sum_{t,m=1}^{T,M}\sum_{l\in\mathbb{L}_{\mathrm{MA}}}h_{n,k,t-l}\tilde{g}_{nlm}\tilde{d}_{ftm}\tilde{y}_{ftm}^{-2}}{\sum_{t,m=1}^{T,M}\sum_{l\in\mathbb{L}_{\mathrm{MA}}}h_{n,k,t-l}\tilde{g}_{nlm}\tilde{y}_{ftm}^{-1}}}, \tag{5.45}$$

$$h_{nkt} \leftarrow h_{nkt}\sqrt{\frac{\sum_{f,m=1}^{F,M}\sum_{l\in\mathbb{L}_{\mathrm{MA}}}w_{nkf}\tilde{g}_{nlm}\tilde{d}_{f,t+l,m}\tilde{y}_{f,t+l,m}^{-2}}{\sum_{f,m=1}^{F,M}\sum_{l\in\mathbb{L}_{\mathrm{MA}}}w_{nkf}\tilde{g}_{nlm}\tilde{y}_{f,t+l,m}^{-1}}}, \tag{5.46}$$

$$\tilde{g}_{nlm} \leftarrow \tilde{g}_{nlm}\sqrt{\frac{\sum_{f,t,k=1}^{F,T,K}w_{nkf}h_{n,k,t-l}\tilde{d}_{ftm}\tilde{y}_{ftm}^{-2}}{\sum_{f,t,k=1}^{F,T,K}w_{nkf}h_{n,k,t-l}\tilde{y}_{ftm}^{-1}}}. \tag{5.47}$$

$\mathbf{B}_{fl}$ depends on only the first term of Eq. (5.36), and $\sum_{l'\in\mathbb{L}_{\mathrm{AR}}}\mathbf{B}_{fl'}\mathbf{x}_{f,t-l'}$ can be rewritten as follows:

$$\sum_{l'\in\mathbb{L}_{\mathrm{AR}}}\mathbf{B}_{fl'}\mathbf{x}_{f,t-l'} = \bar{\mathbf{X}}_{ft}\bar{\mathbf{b}}_f. \tag{5.48}$$

$\bar{\mathbf{b}}_f$ and $\bar{\mathbf{X}}_{ft}$ are given by

$$\bar{\mathbf{b}}_f \triangleq [\mathbf{b}_{f:1}^{\mathsf{T}}, \ldots, \mathbf{b}_{f:M}^{\mathsf{T}}]^{\mathsf{T}} \in \mathbb{C}^{M^2 L_{\mathrm{AR}}} \tag{5.49}$$

$$\mathbf{b}_{f:m} \triangleq [\mathbf{b}_{f,\Delta,m}^{\mathsf{T}}, \ldots, \mathbf{b}_{f,\Delta+L_{\mathrm{AR}}-1,m}^{\mathsf{T}}]^{\mathsf{T}} \in \mathbb{C}^{ML_{\mathrm{AR}}} \tag{5.50}$$

$$\bar{\mathbf{X}}_{ft} \triangleq \mathbf{I}_M \otimes \bar{\mathbf{x}}_{ft}^{\mathsf{T}} \in \mathbb{C}^{M \times M^2 L_{\mathrm{AR}}}, \tag{5.51}$$

$$\bar{\mathbf{x}}_{ft} \triangleq [\mathbf{x}_{f,t-\Delta}^{\mathsf{T}}, \ldots, \mathbf{x}_{f,t-\Delta-L_{\mathrm{AR}}+1}^{\mathsf{T}}]^{\mathsf{T}} \in \mathbb{C}^{ML_{\mathrm{AR}}}, \tag{5.52}$$

where $\mathbf{b}_{flm}$ is the m-th row vector of $\mathbf{B}_{fl}$. Substituting Eq. (5.48) into Eq. (5.36) and letting the partial derivative of Eq. (5.36) with respect to $\bar{\mathbf{b}}_f$ equal to zero, the update rule for $\bar{\mathbf{b}}_f$ is given by

$$\bar{\mathbf{b}}_f = \left(\sum_{t=1}^{T} \bar{\mathbf{X}}_{ft}^{\mathsf{H}} \mathbf{Y}_{ft}^{-1} \bar{\mathbf{X}}_{ft} \right)^{-1} \left(\sum_{t=1}^{T} \bar{\mathbf{X}}_{ft}^{\mathsf{H}} \mathbf{Y}_{ft}^{-1} \mathbf{x}_{ft} \right), \tag{5.53}$$

$$\mathbf{Y}_{ft}^{-1} = \sum_{m=1}^{M} \frac{\mathbf{q}_{fm} \mathbf{q}_{fm}^{\mathsf{H}}}{\tilde{y}_{ftm}}. \tag{5.54}$$

Since the first term accumulates T matrices of size $M^2 L_{\mathrm{AR}} \times M^2 L_{\mathrm{AR}}$, it requires a huge memory and its computational cost is quite expensive. To reduce memory usage and computational cost, the joint optimization method of $\mathbf{Q}_f$ and $\mathbf{B}$, which was initially proposed for AR-ILRMA in [148], is applicable. First, $\mathbf{q}_{fm}^{\mathsf{H}} \bar{\mathbf{d}}_{ft}$ is rewritten by using $\check{\mathbf{x}}_{ft} \triangleq [\mathbf{x}_{ft}^{\mathsf{T}}, \bar{\mathbf{x}}_{ft}^{\mathsf{T}}]^{\mathsf{T}} \in \mathbb{C}^{M(L_{\mathrm{AR}}+1)}$ as follows:

$$\mathbf{q}_{fm}^{\mathsf{H}} \bar{\mathbf{d}}_{ft} = \mathbf{q}_{fm}^{\mathsf{H}} \left(\mathbf{x}_{ft} - \sum_{l' \in \mathbb{L}_{\mathrm{AR}}} \mathbf{B}_{fl'} \mathbf{x}_{f,t-l'}) \right) \tag{5.55}$$

$$= [\mathbf{q}_{fm}^{\mathsf{H}}, -\mathbf{q}_{fm}^{\mathsf{H}} \mathbf{B}_{f,\Delta}, \ldots, -\mathbf{q}_{fm}^{\mathsf{H}} \mathbf{B}_{f,\Delta+L_{\mathrm{AR}}-1}] \check{\mathbf{x}}_{ft} \tag{5.56}$$

$$\triangleq \mathbf{p}_{fm}^{\mathsf{H}} \check{\mathbf{x}}_{ft} \triangleq \mathbf{e}_m^{\mathsf{T}} \mathbf{P}_f \check{\mathbf{x}}_{ft}, \tag{5.57}$$

where $\mathbf{P}_f \triangleq [\mathbf{Q}_f, -\mathbf{Q}_f \mathbf{B}_{f,\Delta}, \ldots, -\mathbf{Q}_f \mathbf{B}_{f,\Delta+L_{\mathrm{AR}}-1}] \triangleq [\mathbf{p}_{f1}, \ldots, \mathbf{p}_{fM}]^{\mathsf{H}} \in \mathbb{C}^{M \times M(L_{\mathrm{AR}}+1)}$. Then, $\mathbf{p}_{fm}$ is updated as follows:

$$\mathbf{c}_{fm} \triangleq \left((\mathbf{Q}_f^{-1} \mathbf{e}_m)^{\mathsf{T}}, \mathbf{0}_{ML_{\mathrm{AR}}}^{\mathsf{T}} \right)^{\mathsf{T}} = \left(\mathbf{u}_{fm}^{\mathsf{T}}, \mathbf{0}_{ML_{\mathrm{AR}}}^{\mathsf{T}} \right)^{\mathsf{T}} \in \mathbb{C}^{M(L_{\mathrm{AR}}+1)}, \tag{5.58}$$

$$\mathbf{\Phi}_{fm} \triangleq \sum_{t=1}^{T} \frac{\check{\mathbf{x}}_{ft} \check{\mathbf{x}}_{ft}^{\mathsf{H}}}{\tilde{y}_{ftm}} \in \mathbb{C}^{M(L_{\mathrm{AR}}+1) \times M(L_{\mathrm{AR}}+1)}, \tag{5.59}$$

$$\mathbf{p}_{fm} \leftarrow \mathbf{\Phi}_{fm}^{-1} \mathbf{c}_{fm} (\mathbf{c}_{fm}^{\mathsf{H}} \mathbf{\Phi}_{fm}^{-1} \mathbf{c}_{fm})^{-\frac{1}{2}}, \tag{5.60}$$

where $\mathbf{u}_{fm}$ is the m-th column vector of $\mathbf{U}_f \triangleq \mathbf{Q}_f^{-1}$. In our experiment in Section 5.5, Eq. (5.60) was used for updating $\mathbf{Q}$ and $\mathbf{B}$.

To satisfy the normalization constraints given by Eqs. (5.32), (5.33), and (5.34), we adjust the scales of $\mathbf{Q}$, $\tilde{\mathbf{G}}$, and $\mathbf{W}$ in this order in each iteration by using

$$\mu_f \triangleq \frac{1}{M}\mathrm{tr}(\mathbf{Q}_f\mathbf{Q}_f^{\mathsf{H}}), \qquad \begin{cases}\mathbf{P}_f \leftarrow \mu_f^{-\frac{1}{2}}\mathbf{P}_f,\\ w_{nkf} \leftarrow \mu_f^{-1}w_{nkf},\end{cases} \tag{5.61}$$

$$\phi_n \triangleq \sum_{l\in\mathbb{L}_{\mathrm{MA}}}\sum_{m=1}^{M}\tilde{g}_{nlm}, \quad \begin{cases}\tilde{g}_{nlm} \leftarrow \phi_n^{-1}\tilde{g}_{nlm},\\ w_{nkf} \leftarrow \phi_n w_{nkf},\end{cases} \tag{5.62}$$

$$\nu_{nk} \triangleq \sum_{f=1}^{F} w_{nkf} \qquad \begin{cases}w_{nkf} \leftarrow \nu_{nk}^{-1}w_{nkf},\\ h_{nkt} \leftarrow \nu_{nk}h_{nkt}.\end{cases} \tag{5.63}$$

5.3.3 Rank-Constrained Extension

If the similar spectra continue for a few time frames ($\{\lambda_{nft}\}_{f=1}^{F} \approx \{\lambda_{n,f,t-l}\}_{f=1}^{F}$) and the SCM of the direct signal, $\mathbf{G}_{nf0}$, is close to the SCM of the early reflection, $\mathbf{G}_{n,f,l(>0)}$, the distribution of the direct signal can be close to that of the early reflection as follows:

$$\mathcal{N}_{\mathbb{C}}\left(\mathbf{0}_M, \lambda_{nft}\mathbf{G}_{nf0}\right) \approx \mathcal{N}_{\mathbb{C}}\left(\mathbf{0}_M, \lambda_{n,f,t-l}\mathbf{G}_{nfl}\right). \tag{5.64}$$

As a result, a part of the direct signal can be included in the early reflection.

When $\tilde{\mathbf{g}}_{n0}$ is initialized with $[\epsilon, \ldots, \epsilon, 1, \epsilon, \ldots, \epsilon]$, where the n-th element is one, $\mathbf{G}_{nf0}$ mainly consists of the n-th rank-1 matrix given by $\mathbf{u}_{fn}\mathbf{u}_{fn}^{\mathsf{H}}$, where $\mathbf{u}_{fn}$ is the n-th column vector of $\mathbf{Q}_f^{-1}$. To keep $\mathbf{G}_{n,f,l(>0)}$ away from $\mathbf{G}_{nf0}$, we restricts $\mathbf{G}_{n,f,l(>0)}$ by setting $\tilde{\mathbf{g}}_{n,l(>0)}$ to $[\epsilon, \ldots, \epsilon, 0, \epsilon, \ldots, \epsilon]$, where the n-th element is zero. Because of the multiplicative update rule of $\tilde{\mathbf{g}}_{nl}$ given by Eq. (5.47), once $\tilde{g}_{nlm}$ is set to zero, it is kept to zero and the rank of $\mathbf{G}_{n,f,l(>0)}$ is kept to $M-1$.

5.4 FastMNMF2 with an ARMA Model and a Deep Speech Prior (ARMA-FastMNMF2-DSP)

ARMA-FastMNMF2-DSP can be derived by replacing NMF-based source model for one of sources with DNN-based speech model as in FastMNMF-DSP.

5.4.1 Formulation

We assume that the observed signals include only one speech and N' ($\triangleq N-1$) noise (N sources in total), and source 0 corresponds to speech and source $n\,(1 \leq n \leq N')$ corresponds to noise. Now $\{\lambda_{nft}\}_{n=0}^{N'}$ are given by

$$\lambda_{nft} = \begin{cases} u_f v_t [\boldsymbol{\sigma}^2_{\boldsymbol{\theta}}(\mathbf{z}_t)]_f & (n=0), \\ \displaystyle\sum_{k=1}^{K} w_{nkf} h_{nkt} & (1 \leq n \leq N'). \end{cases} \tag{5.65}$$

The log-likelihood function of ARMA-FastMNMF2-DSP is obtained by substituting $\tilde{y}_{ftm} = \sum_{l \in \mathbb{L}_{\mathrm{MA}}} u_f v_t [\boldsymbol{\sigma}^2_{\boldsymbol{\theta}}(\mathbf{z}_{t-l})]_f \tilde{g}_{0ml} + \sum_{n,k=1}^{N,K} \sum_{l \in \mathbb{L}_{\mathrm{MA}}} w_{nkf} h_{nkt} \tilde{g}_{nml}$ into Eq. (5.38). To avoid the scale ambiguity, we put the normalization constraints given by Eqs. (5.32), (5.33), and (5.34) and

$$\sum_{f=1}^{F} u_f = 1 \tag{5.66}$$

5.4.2 Optimization

To update the latent variables $\mathbf{Z}$ included in Eq. (5.65), we use backpropagation such that the log-likelihood $\log p(\mathbf{X}|\mathbf{W}, \mathbf{H}, \mathbf{U}, \mathbf{V}, \mathbf{Z}, \tilde{\mathbf{G}}, \mathbf{Q}, \mathbf{B})$ is maximized with respect to $\mathbf{z}_t$. As in the NMF-based source model, the MU rules of $\mathbf{U}$ and $\mathbf{V}$ are given by

$$u_f \leftarrow u_f \sqrt{\frac{\sum_{t,m=1}^{T,M} \sum_{l \in \mathbb{L}_{\mathrm{MA}}} v_{t-l} [\boldsymbol{\sigma}^2_{\boldsymbol{\theta}}(\mathbf{z}_{t-l})]_f \tilde{g}_{0lm} \tilde{d}_{ftm} \tilde{y}^{-2}_{ftm}}{\sum_{t,m=1}^{T,M} \sum_{l \in \mathbb{L}_{\mathrm{MA}}} v_{t-l} [\boldsymbol{\sigma}^2_{\boldsymbol{\theta}}(\mathbf{z}_{t-l})]_f \tilde{g}_{0lm} \tilde{y}^{-1}_{ftm}}}, \tag{5.67}$$

$$v_t \leftarrow v_t \sqrt{\frac{\sum_{f,m=1}^{F,M} \sum_{l \in \mathbb{L}_{\mathrm{MA}}} u_f [\boldsymbol{\sigma}^2_{\boldsymbol{\theta}}(\mathbf{z}_t)]_f \tilde{g}_{0lm} \tilde{d}_{f,t+l,m} \tilde{y}^{-2}_{f,t+l,m}}{\sum_{f,m=1}^{F,M} \sum_{l \in \mathbb{L}_{\mathrm{MA}}} u_f [\boldsymbol{\sigma}^2_{\boldsymbol{\theta}}(\mathbf{z}_t)]_f \tilde{g}_{0lm} \tilde{y}^{-1}_{f,t+l,m}}} \tag{5.68}$$

$\mathbf{W}$, $\mathbf{H}$, $\tilde{\mathbf{G}}$, $\mathbf{Q}$, and $\mathbf{B}$ are updated in the same way as FastMNMF1 using Eqs. (5.45), (5.46), (5.47), and (5.60).

To satisfy the normalization constraints given by Eqs. (5.32), (5.33), (5.34), and (5.66), we adjust the scales of $\mathbf{Q}$, $\tilde{\mathbf{G}}$, $\mathbf{U}$, and $\mathbf{W}$ in this order in each iteration by

using

$$\mu_f \triangleq \frac{1}{M}\mathrm{tr}(\mathbf{Q}_f\mathbf{Q}_f^{\mathsf{H}}), \quad \begin{cases} \mathbf{P}_f \leftarrow \mu_f^{-\frac{1}{2}}\mathbf{P}_f, \\ u_f \leftarrow \mu_f^{-1}u_f, \\ w_{nkf} \leftarrow \mu_f^{-1}w_{nkf} \quad (1 \le n \le N'), \end{cases} \tag{5.69}$$

$$\phi_n \triangleq \sum_{l\in\mathbb{L}_{\mathrm{MA}}}\sum_{m=1}^{M}\tilde{g}_{nlm}, \quad \begin{cases} \tilde{g}_{nlm} \leftarrow \phi_n^{-1}\tilde{g}_{nlm}, \\ u_f \leftarrow \phi_1 u_f, \\ w_{nkf} \leftarrow \phi_n w_{nkf} \quad (1 \le n \le N'), \end{cases} \tag{5.70}$$

$$\psi \triangleq \sum_{f=1}^{F} u_f, \quad \begin{cases} u_f \leftarrow \psi^{-1}u_f \\ v_t \leftarrow \psi v_t, \end{cases} \tag{5.71}$$

and Eq. (5.63).

5.5 Evaluation

This section reports comparative experiments conducted for evaluating the effectiveness of ARMA-FastMNMF2. To draw the full potential of ARMA-FastMNMF2, first, we comprehensively investigated the configuration of M, K, L_{MA}, L_{AR}, and Δ. Then, we tested ARMA-FastMNMF2-DSP for speech enhancement. Through all experiments, audio signals were sampled at 16 kHz and processed by STFT with a shifting interval of 256 points and a Hann window of 1024 points (F= 513).

5.5.1 Comparison of Model Complexities

In the speech separation task, we comprehensively investigated the SDRs obtained by ARMA-FastMNMF2 with different complexities.

Experimental Conditions

We prepared a dataset of eight channel reverberant mixture signals using the simulation data of REVERB Challenge dataset [149]. Each mixture signal consisted of two reverberant speech signals synthesized by convolving dry speech signals with real impulse responses from the development and evaluation subsets of REVERB Challenge dataset. The impulse responses were recorded in three rooms

with the reverberation times RT_{60} of 250 ms, 500 ms, and 700 ms. The distances between sound sources and microphones were set to 0.5 m (near) and 2.0 m (far). We thus tested six conditions in total, where 20 signals were used for each condition. Audio signals were sampled at 16 kHz and processed by STFT with a Hann window of 1024 points ($F = 513$) and a shifting interval of 256 points.

We tested ARMA-FastMNMF2 with $M \in \{3, 8\}$ microphones, $N = 2$ sources, $K = 16$ bases, $L_{\mathrm{MA}} \in \{0, 2, 4, 8, 16\}$, and $L_{\mathrm{AR}} \in \{0, 2, 4, 8, 16\}$. ARMA-FastMNMF2 with $L_{\mathrm{MA}} = 0$ and $L_{\mathrm{AR}} = 0$ is equivalent to vanilla FastMNMF2, one with $L_{\mathrm{MA}} \neq 0$ and $L_{\mathrm{AR}} = 0$ is equivalent to MA-FastMNMF2, and one with $L_{\mathrm{MA}} = 0$ and $L_{\mathrm{AR}} \neq 0$ is equivalent to AR-FastMNMF2. We tested $\Delta \in \{2, 3, 4\}$ for AR-FastMNMF2. For ARMA-FastMNMF2, we decided to use $\Delta = 2$ when $M = 3$ and $\Delta = 3$ when $M = 8$ based on the results of AR-FastMNMF2. To draw the full potential, $\mathbf{Q}$ and $\tilde{\mathbf{G}}$ of MA-FastMNMF2 were initialized to those estimated by FastMNMF2 with $K = 2$ and 50 iterations as the gradual initialization method described in Section 4.5.4. $\mathbf{Q}$, $\tilde{\mathbf{G}}$, and $\mathbf{B}$ of AR-FastMNMF2 and ARMA-FastMNMF2 were initialized to those estimated by AR-FastMNMF2 with $K = 2$ and 50 iterations. $\mathbf{W}$ and $\mathbf{H}$ were initialized randomly. MA-FastMNMF2, AR-FastMNMF2, and ARMA-FastMNMF2 were then updated 100 times.

We used the signal-to-distortion ratio (SDR) [136, 137] for evaluating the source estimation and dereverberation performance. Dry speech signals without reverberation were used as reference signals. The elapsed time per iteration for processing a 3-s signal was measured on Intel Xeon W-2145 (3.70 GHz).

Experimental Results

Table. 5.1 shows the elapsed times per iteration for processing a 3-s signal on CPU. $L_{\mathrm{MA}} = 0$ and $L_{\mathrm{AR}} = 0$ corresponds to vanilla FastMNMF2, $L_{\mathrm{MA}} \neq 0$ and $L_{\mathrm{AR}} = 0$ corresponds to MA-FastMNMF2, and $L_{\mathrm{MA}} = 0$ and $L_{\mathrm{AR}} \neq 0$ corresponds to AR-FastMNMF2. When $M = 3$, the computational cost for estimating the AR filters was equivalent to that for estimating the MA filters. However, when $M = 8$, the computation of the AR filters took much longer time especially when L_{AR} was large.

Table 5.1: The elapsed times [sec] per iteration for processing a 3-s signal on CPU (Intel Xeon W-2145 3.70 GHz).

L_{MA} \ L_{AR}	**0**	**2**	**4**	**8**	**16**
0	0.406	0.537	0.951	2.14	6.58
2	0.824	0.974	1.26	2.59	6.99
4	1.24	1.40	1.67	2.99	7.39
8	2.06	2.23	2.57	3.80	8.24
16	3.63	3.82	4.07	5.42	9.76

(a) $M = 3$

L_{MA} \ L_{AR}	**0**	**2**	**4**	**8**	**16**
0	0.962	4.52	11.9	36.4	115
2	1.69	5.27	12.8	37.4	115
4	2.43	6.05	13.4	38.0	116
8	3.87	7.50	14.9	39.9	117
16	6.71	10.4	17.7	42.5	120

(b) $M = 8$

Table. 5.2 shows the SDRs of FastMNMF2 averaged over the 20 mixtures for each condition. In four conditions out of six conditions, FastMNMF2 with $M = 3$ outperformed one with $M = 8$. This is probably because reverberations make it difficult to estimate a larger number of parameters of FastMNMF2 with $M = 8$.

Tables. 5.3 and 5.4 show the average SDRs of MA-FastMNMF2 with $M = 3$ and $M = 8$, respectively. When $\mathrm{RT}_{60} = 500$ms or 700ms, longer L_{MA} achieved better performances, and the SDR improvements compared to vanilla FastMNMF2 with $M = 3$ were more than 1.8 dB. The same tendency was observed in MA-FastMNMF with $M = 8$, but, in far conditions with $\mathrm{RT}_{60} = 500$ms or 700ms, the SDR improvements compared to vanilla FastMNMF2 were more than 3.9 dB. Since the joint-diagonalization constraint given by Eq. (5.28) indicates that $N(L_{\mathrm{MA}} + 1)$ SCMs consist of M rank-1 matrices, when M is small, this constraint strongly restricts SCMs, resulting in the limited performances.

Figs. 5.2 and 5.3 show the average SDRs of AR-FastMNMF2 with $M = 3$ and $M = 8$, respectively. In all cases except for the near condition with $M = 8$ and $\mathrm{RT}_{60} = 250$ ms, AR-FastMNMF2 with the best parameters outperformed MA-FastMNMF2, especially when the reverberation time was long, because the AR model is suitable for representing the long reverberations. One drawback of AR-FastMNMF2 is the sensitivity to L_{AR}. When L_{AR} was too long compared to the actual reverberations, its performance drastically degraded as in the near condition with $M = 8$ and $\mathrm{RT}_{60} = 250$ ms.

Table 5.2: The average SDRs [dB] of vanilla FastMNMF2 with $M = 3$.

Distance RT_{60}	far 250ms	far 500ms	far 700ms	near 250ms	near 500ms	near 700ms
$M = 3$	8.8	1.9	2.0	12.4	7.9	8.6
$M = 8$	6.7	2.4	1.3	14.1	5.9	6.8

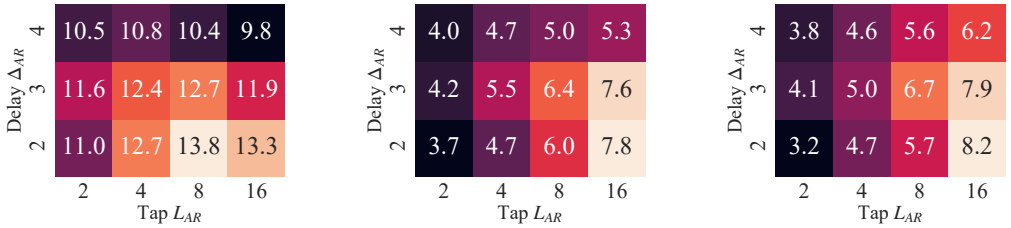
Table 5.3: The average SDRs obtained by MA-FastMNMF2 with $M = 3$.

Distance RT_{60}	far 250ms	far 500ms	far 700ms	near 250ms	near 500ms	near 700ms
$L_{\mathrm{MA}} = 2$	9.7	2.9	2.8	13.2	9.1	10.0
$L_{\mathrm{MA}} = 4$	9.6	3.2	3.2	13.3	9.4	10.4
$L_{\mathrm{MA}} = 8$	9.4	3.5	3.7	13.4	9.6	10.5
$L_{\mathrm{MA}} = 16$	9.4	3.7	4.0	13.3	9.6	10.6

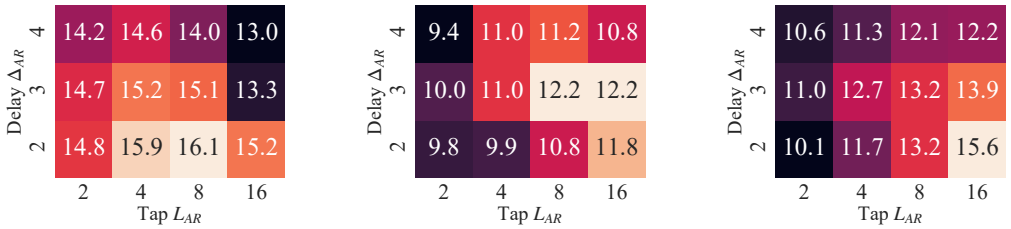
Table 5.4: The average SDRs obtained by MA-FastMNMF2 with $M = 8$.

Distance RT_{60}	far 250ms	far 500ms	far 700ms	near 250ms	near 500ms	near 700ms
$L_{\mathrm{MA}} = 2$	8.0	3.8	2.8	16.1	7.8	9.2
$L_{\mathrm{MA}} = 4$	8.3	5.0	3.9	16.6	8.4	9.8
$L_{\mathrm{MA}} = 8$	8.6	5.9	4.9	16.7	8.6	10.3
$L_{\mathrm{MA}} = 16$	8.6	6.3	5.3	16.8	8.6	10.5

Figs. 5.4 and 5.5 show the average SDRs of ARMA-FastMNMF2 with $M = 3$ and $M = 8$, respectively. In all cases, ARMA-FastMNMF2 outperformed AR-FastMNMF2 with the same Δ ($\Delta = 2$ when $M = 3$ and $\Delta = 3$ when $M = 8$), although the SDR improvements from AR-FastMNMF2 was small when $M = 3$ because of the joint diagonalization constraint in the MA model. When $M = 8$, one advantage of ARMA-FastMNMF2 compared to AR-FastMNMF2 is that ARMA-FastMNMF2 with small L_{AR} and large L_{MA} worked as well as AR-FastMNMF2 with larger L_{AR}. For example, in the far condition with $\mathrm{RT}_{60} = 500$ ms, ARMA-FastMNMF2 with $L_{\mathrm{AR}} = 2$ and $L_{\mathrm{MA}} = 16$ outperformed AR-FastMNMF2 with $L_{\mathrm{AR}} = 8$. Thus, ARMA-FastMNMF2 often achieved better performance with lower computational cost as shown in Table 5.1b.

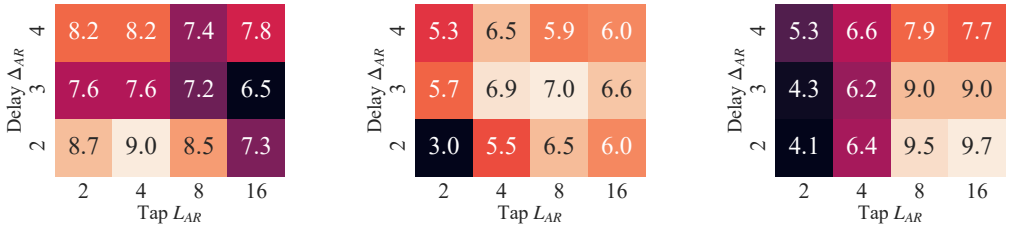


(a) far $RT_{60} = 250$ms (b) far $RT_{60} = 500$ms (c) far $RT_{60} = 700$ms

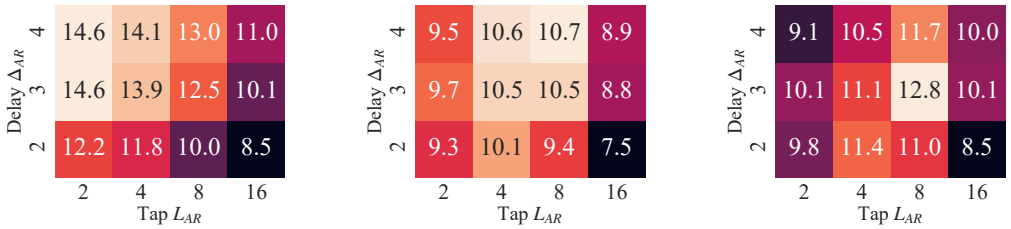


(d) near $RT_{60} = 250$ms (e) near $RT_{60} = 500$ms (f) near $RT_{60} = 700$ms

Figure 5.2: The average SDRs obtained by AR-FastMNMF2 with $M = 3$.

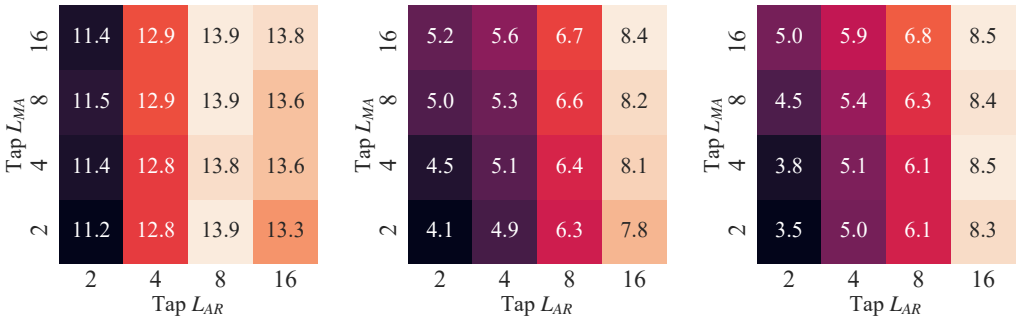


(a) far $RT_{60} = 250$ms (b) far $RT_{60} = 500$ms (c) far $RT_{60} = 700$ms

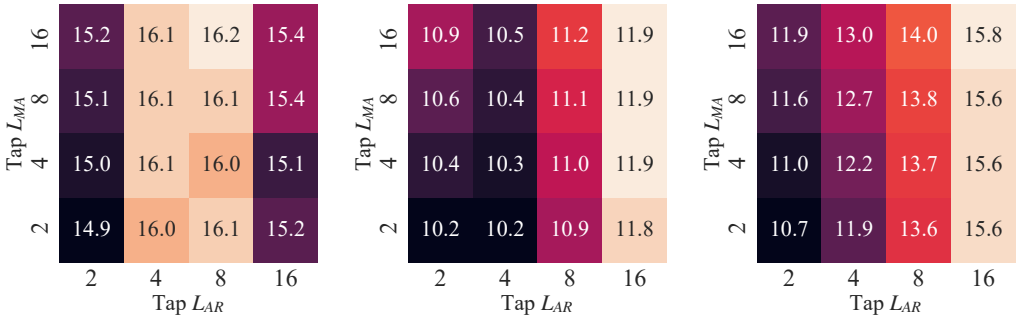


(d) near $RT_{60} = 250$ms (e) near $RT_{60} = 500$ms (f) near $RT_{60} = 700$ms

Figure 5.3: The average SDRs obtained by AR-FastMNMF2 with $M = 8$.

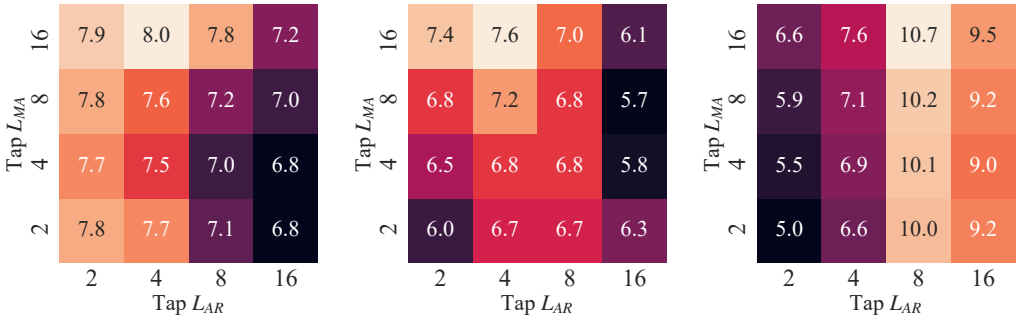


(a) far $RT_{60} = 250$ms (b) far $RT_{60} = 500$ms (c) far $RT_{60} = 700$ms

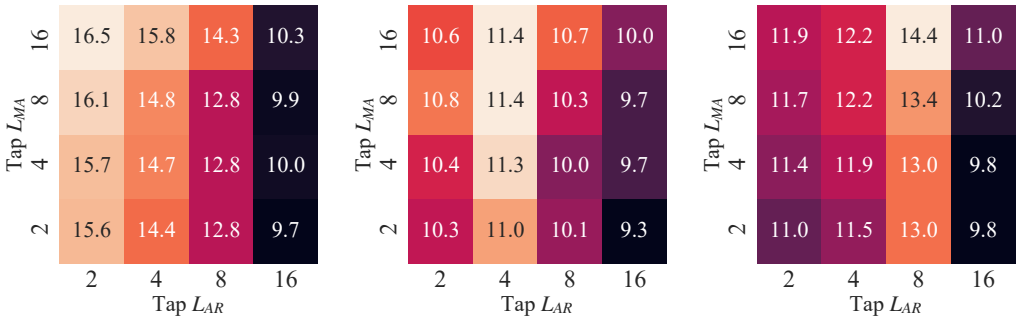


(d) near $RT_{60} = 250$ms (e) near $RT_{60} = 500$ms (f) near $RT_{60} = 700$ms

Figure 5.4: The average SDRs obtained by ARMA-FastMNMF2 with $M = 3$.



(a) far $RT_{60} = 250$ms (b) far $RT_{60} = 500$ms (c) far $RT_{60} = 700$ms



(d) near $RT_{60} = 250$ms (e) near $RT_{60} = 500$ms (f) near $RT_{60} = 700$ms

Figure 5.5: The average SDRs obtained by ARMA-FastMNMF2 with $M = 8$.

5.5.2 Comparison with the State-of-the-Art BSS Methods in Speech Separation and Denoising on Simulated Data

We compared ARMA-FastMNMF2 with the conventional BSS methods and sequential methods in separation of reverberant noisy speech mixtures.

Experimental Conditions

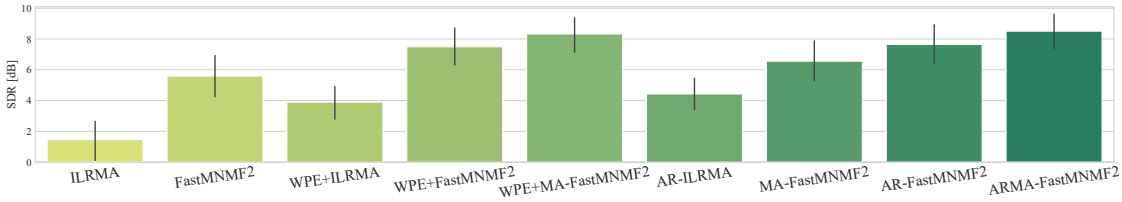
We prepared a dataset of eight channel noisy reverberant mixture signals using the simulation data of REVERB Challenge dataset [149]. Each mixture signal consisted of diffuse noise recorded in real environments and two reverberant speech signals synthesized by convolving dry speech signals with real impulse responses from the development and evaluation subsets of REVERB Challenge dataset. The signal-to-noise ratio (SNR) between dry mixture images and noise was set to 0 dB.

For comparison, we tested ILRMA, FastMNMF2, the sequential use of WPE [19, 113] and ILRMA [28], that of WPE and FastMNMF2 [43], that of WPE and MA-FastMNMF2, AR-ILRMA [118], MA-FastMNMF2, AR-FastMNMF2, and ARMA-FastMNMF2. The number of microphone was set to $M \in [3, 8]$, All methods were configured with $N = M$, and $K = 16$. When $M = 3$, Δ was set to 2, and when $M = 8$, Δ was set to 3. The tap length for the MA model was set to $L_{\mathrm{MA}} = 8$, and the tap length for the AR model was set to $L_{\mathrm{MA}} = 4$. In the sequential methods, WPE was updated 10 times, and then ILRMA, FastMNMF2, or MA-FastMNMF2 was updated 150 times. MA-, AR-, and ARMA-FastMNMF2 were initialized by the same way as the previous experiment. Similarly, AR-ILRMA was initialized using AR-ILRMA with $K = 2$. MA-FastMNMF2, AR-FastMNMF2, ARMA-FastMNMF2, and AR-ILRMA were then updated 100 times.

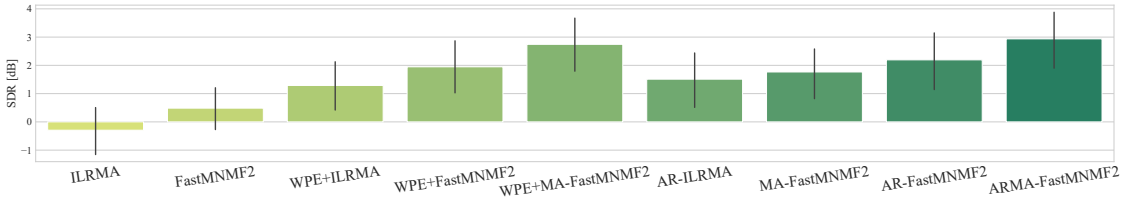
We used the SDR [136, 137] for evaluating the source estimation and dereverberation performance.

Experimental Results

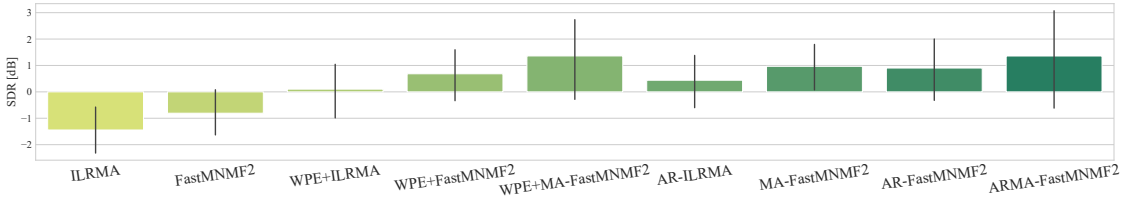
Figs. 5.6 and 5.7 show the average SDRs of the proposed and conventional methods with $M = 3$ and $M = 8$, respectively. For $M = 3$, ARMA-FastMNMF2



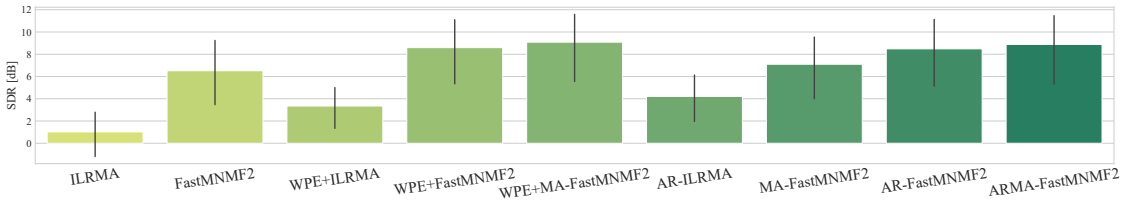
(a) far $\mathrm{RT}_{60} = 250\mathrm{ms}$



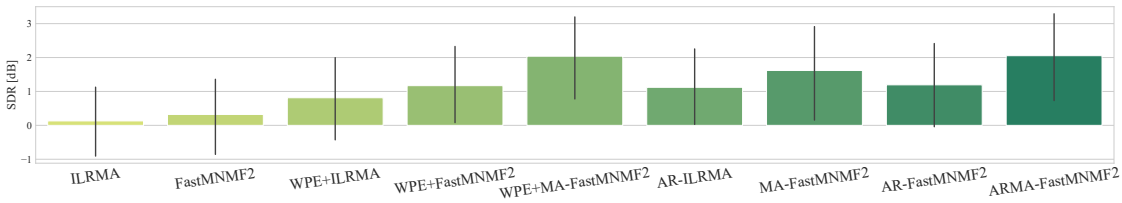
(b) far $\mathrm{RT}_{60} = 500\mathrm{ms}$



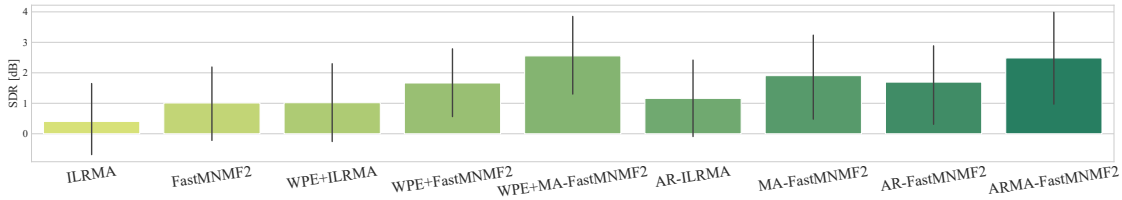
(c) far $\mathrm{RT}_{60} = 700\mathrm{ms}$



(d) near $\mathrm{RT}_{60} = 250\mathrm{ms}$

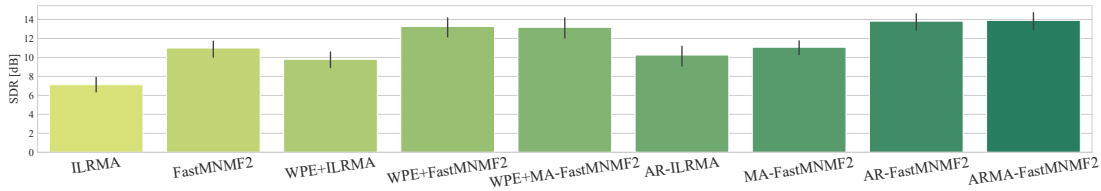


(e) near $\mathrm{RT}_{60} = 500\mathrm{ms}$

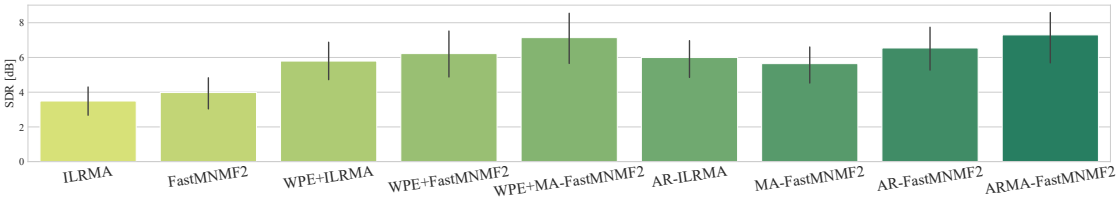


(f) near $\mathrm{RT}_{60} = 700\mathrm{ms}$

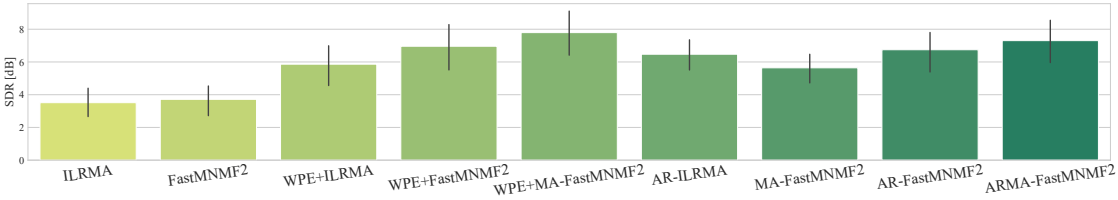
Figure 5.6: The average SDRs in speech separation of simulated data ($M = 3$).



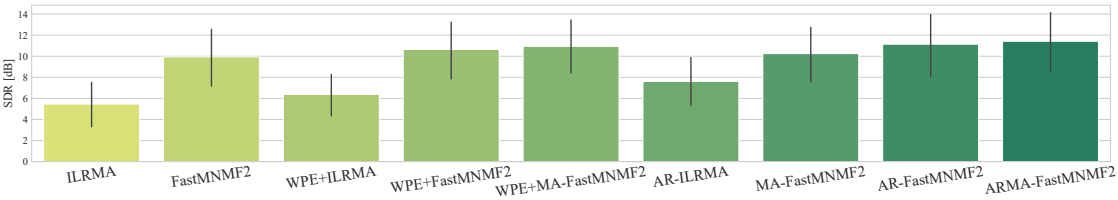
(a) far $\mathrm{RT}_{60} = 250$ms



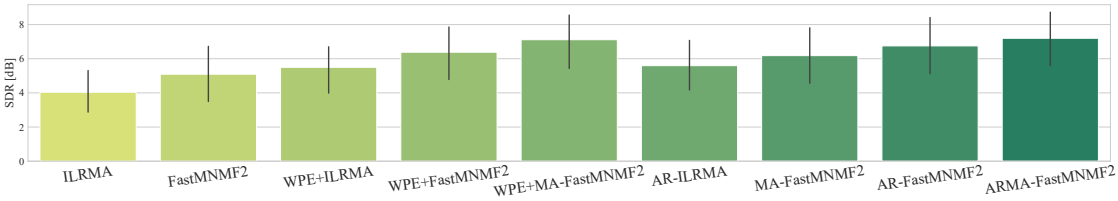
(b) far $\mathrm{RT}_{60} = 500$ms



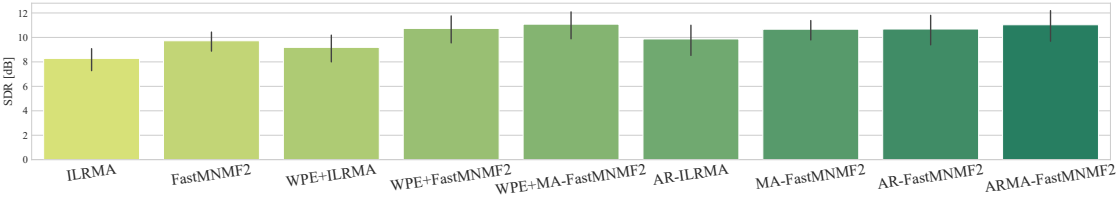
(c) far $\mathrm{RT}_{60} = 700$ms



(d) near $\mathrm{RT}_{60} = 250$ms



(e) near $\mathrm{RT}_{60} = 500$ms



(f) near $\mathrm{RT}_{60} = 700$ms

Figure 5.7: The average SDRs in speech separation of simulated data ($M = 8$).

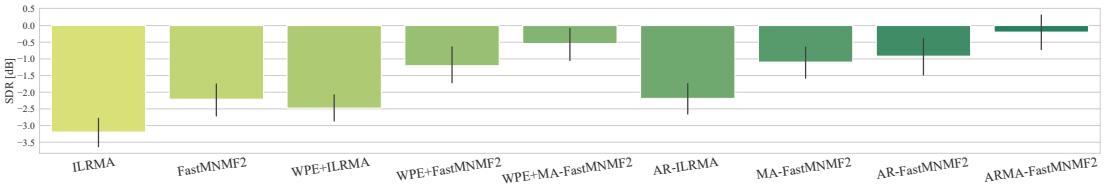
(4.8 dB on average) and WPE+MA-FastMNMF2 (4.8 dB) outperformed ILRMA (0.2 dB), FastMNMF2 (2.6 dB), WPE+ILRMA (2.4 dB), WPE+FastMNMF2 (4.0 dB), AR-ILRMA (2.7 dB), MA-FastMNMF2 (3.3 dB), and AR-FastMNMF2 (4.1 dB). For $M = 8$, ARMA-FastMNMF2 (9.7 dB on average) outperformed ILRMA (5.3 dB), FastMNMF2 (7.2 dB), WPE+ILRMA (7.1 dB), WPE+FastMNMF2 (9.0 dB), AR-ILRMA (7.6 dB), MA-FastMNMF2 (8.2 dB), AR-FastMNMF2 (9.3 dB), and WPE+MA-FastMNMF2 (9.5 dB). Since the methods based on FastMNMF2 can deal with diffuse noise because of the JD full-rank spatial model, the performances of MA-, AR-, and ARMA-FastMNMF2 were better than those of AR-ILRMA and WPE+ILRMA based on the rank-1 spatial model. In this experiment, we showed the effectiveness of jointly optimize the MA model-based dereverberation and source separation. However, the effectiveness of jointly optimize the AR model-based dereverberation and source separation was very small. This was probably because of the difficulties of jointly optimize a large number of parameters. One possible way to alleviate this problem is to restrict the SCMs of the direct signals to rank-1 matrices as in rank-constrained FastMNMF described in Section 4.6 to reduce the degree of freedom.

5.5.3 Comparison with the State-of-the-Art BSS Methods in Speech Separation and Denoising on Real Data

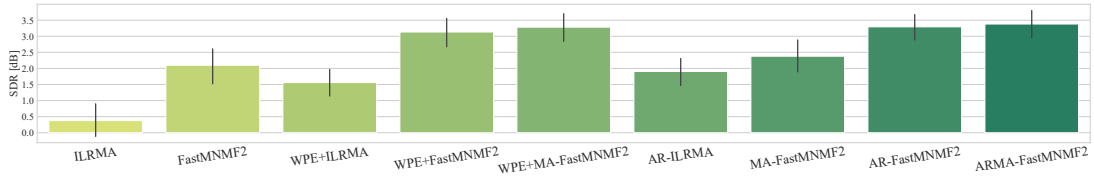
We compared the proposed ARMA-FastMNMF2 using a dataset recorded in a real environment.

Experimental Conditions

We used almost the same dataset as the one used in Section 4.7.8. An eight-channel microphone array ($M = 8$) and three loudspeakers corresponding to two speech sources and one noise source were put in a spacious, heavily-echoic room with $\mathrm{RT}_{60} = 800$ ms. We randomly selected 20 clean speech signals from the WSJ-0 corpus and four noise signals from the CHiME3 evaluation dataset. To obtain ground-truth images, these signals were recorded individually and 50 mixtures were synthesized by superimposing randomly-selected speech and



(a) $M = 3$



(b) $M = 8$

Figure 5.8: The average SDRs in speech separation of real recordings.

noise signals, where the signal-to-noise ratio (SNR) was set to 0 dB. The number of microphone was set to $M \in [3, 8]$. We tested the same methods with the same configurations as the experiment in Section 5.5.2. We used the SDR [136, 137] for evaluating the source separation and dereverberation performance.

Experimental Results

Figs. 5.8(a) and 5.8(b) show the average SDRs of the proposed and conventional methods with $M = 3$ and $M = 8$, respectively. We got the similar results as the experiment in Section 5.5.2. For $M = 3$, ARMA-FastMNMF2 (-0.2 dB) outperformed ILRMA (-3.2 dB), FastMNMF2 (-2.2 dB), WPE+ILRMA (-2.5 dB), WPE+FastMNMF2 (-1.2 dB), WPE+MA-FastMNMF2 (-0.5 dB) AR-ILRMA (-2.2 dB), MA-FastMNMF2 (-1.1 dB), AR-FastMNMF2 (-0.9 dB). For $M = 8$, ARMA-FastMNMF2 (3.4 dB) outperformed ILRMA (0.4 dB), FastMNMF2 (2.1 dB), WPE+ILRMA (1.6 dB), WPE+FastMNMF2 (3.1 dB), WPE+MA-FastMNMF2 (3.3 dB), AR-ILRMA (1.9 dB), MA-FastMNMF2 (2.4 dB), and AR-FastMNMF2 (3.3 dB). Since the reverberation time was longer than that of the simulated data used in the experiment in Section 5.5.2 and the noise was more complicated, the SDR of each method was lower than that of the experiment in Section 5.5.2.

The performance of AR-FastMNMF2 and ARMA-FastMNMF2 did not significantly differ, especially when $M = N = 8$. For $M = N = 8$, two out of eight

separated signals of AR-FastMNMF2 included the direct signals, and other two signals often included the reverberations that could not be removed by the AR model. In ARMA-FastMNMF2, the amount of reverberations included in the separated signals that did not correspond to the direct signals was smaller than that of AR-FastMNMF2, because the reverberations that could not be removed by the AR model were removed by the MA model. Since the SDRs were calculated using only the separated signals of the direct signals, the SDRs of AR-FastMNMF2 and ARMA-FastMNMF2 were equivalent. When some of the separated signals include only the reverberations, the estimation of the number of actual sound sources from the eight separated signals is often difficult. The incorrect estimation might result in a severe problem in real applications.

5.5.4 Comparison with the State-of-the-Art BSS Methods in Speech Enhancement

We evaluated the effectiveness of the DNN-based speech model (Section. 5.4) and the rank constraint (Section. 5.3.3) in speech enhancement.

Experimental Conditions

We made eight-channel noisy reverberant signals using the simulation data of REVERB Challenge dataset [149]. Each mixture signal consisted of diffuse noise recorded in a real environment and a reverberant speech signal synthesized by convolving dry speech signals with real impulse response from the evaluation subsets of REVERB Challenge dataset. The signal-to-noise ratio (SNR) between dry images and noise was set to 0 dB.

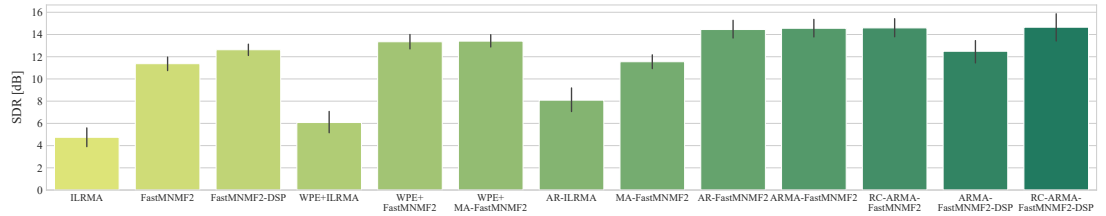
For comparison, in addition to the methods used in Section 5.5.2, we tested FastMNMF2-DSP (Section 4.4), ARMA-FastMNMF2-DSP (Section 5.4), and the rank-constrained version of ARMA-FastMNMF2 and ARMA-FastMNMF2-DSP called RC-ARMA-FastMNMF2 and RC-ARMA-FastMNMF2-DSP. We used the same parameters as those used in Section 5.5.2. The deep speech generative model was trained in the same way as the experiment in Section 3.5. $\mathbf{Q}$, $\tilde{\mathbf{G}}$, and $\mathbf{B}$ of ARMA-FastMNMF2-DSP and RC-ARMA-FastMNMF2-DSP were initially

estimated by ARMA-FastMNMF2 whose $\mathbf{Q}$ was initialized with the eigenvectors of the mixture SCMs and $K = 2$. The latent variables $\mathbf{Z}$ were initially estimated from the source spectra estimated by ARMA-FastMNMF2 by using the decoder of the VAE. FastMNMF2-DSP was also initialized similarly by using FastMNMF2. The other methods were initialized in almost the same way as the experiment in Section 5.5.2 except for $\mathbf{Q}$, which was initialized with the eigenvectors of the mixture SCMs. We used the SDR [136,137] for evaluating the speech enhancement and dereverberation performance.

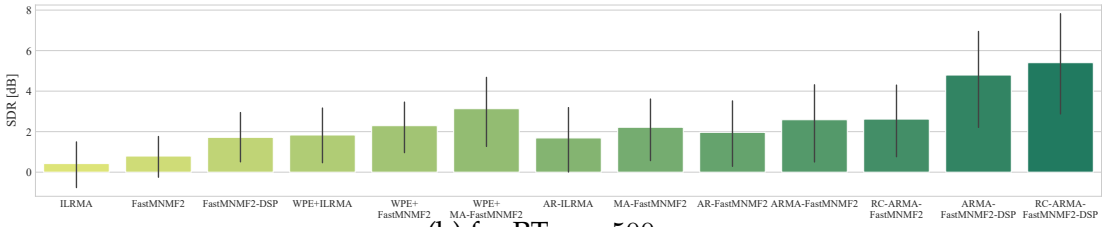
Experimental Results

Figs. 5.9 and 5.10 show the average SDRs of the methods with $M = 3$ and $M = 8$, respectively. As to the methods based on the NMF-based source model, the results were consistent with those in Section 5.5.2. For $M = 3$, ARMA-FastMNMF2 (9.4 dB on average) and WPE+MA-FastMNMF2 (9.4 dB) outperformed ILRMA (3.9 dB), FastMNMF2 (7.1 dB), WPE+ILRMA (5.6 dB), WPE+FastMNMF2 (9.2 dB), AR-ILRMA (6.0 dB), MA-FastMNMF2 (8.3 dB), and AR-FastMNMF2 (8.8 dB). For $M = 8$, ARMA-FastMNMF2 (12.3 dB) and WPE+MA-FastMNMF2 (12.3 dB) outperformed ILRMA (7.7 dB), FastMNMF2 (10.4 dB), WPE+ILRMA (9.3 dB), WPE+FastMNMF2 (12.2 dB), AR-ILRMA (9.6 dB), MA-FastMNMF2 (10.8 dB), and AR-FastMNMF2 (11.8 dB).

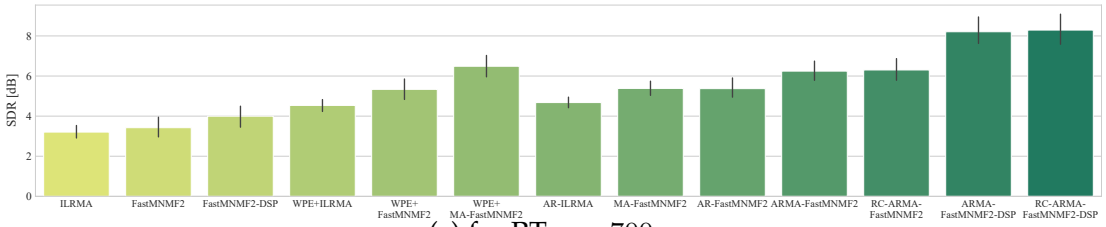
When $\mathrm{RT}_{60} = 500$ ms or 700 ms, ARMA-FastMNMF2-DSP outperformed the other methods by a large margin for $M = 3$, and the gain became small for $M = 8$. In contrast, when $\mathrm{RT}_{60} = 250$ ms, ARMA-FastMNMF2-DSP underperformed the NMF-based methods. In ARMA-FastMNMF2-DSP, we found that not only reverberation but also a part of the direct signal were represented with the MA model, probably because of the reason discussed in Section 5.3.3. Thanks to the rank constraint discussed in Section. 5.3.3, RC-ARMA-FastMNMF2-DSP significantly outperformed ARMA-FastMNMF2-DSP, especially when $\mathrm{RT}_{60} = 250$ ms. For ARMA-FastMNMF2, in contrast, the rank constraint was not effective, because of the limited representation power of the NMF-based source model and the difference of parameter estimation.



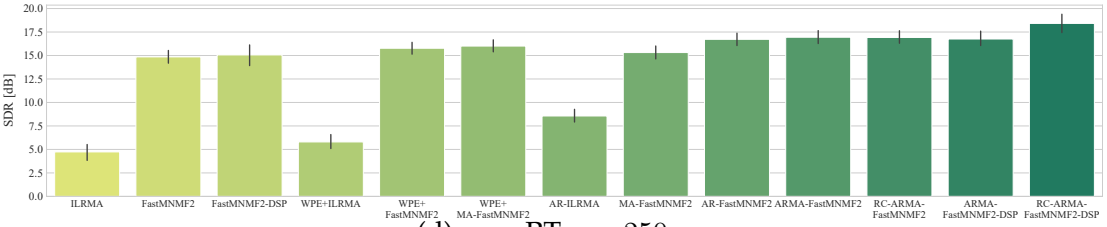
(a) far $\mathrm{RT}_{60} = 250\mathrm{ms}$



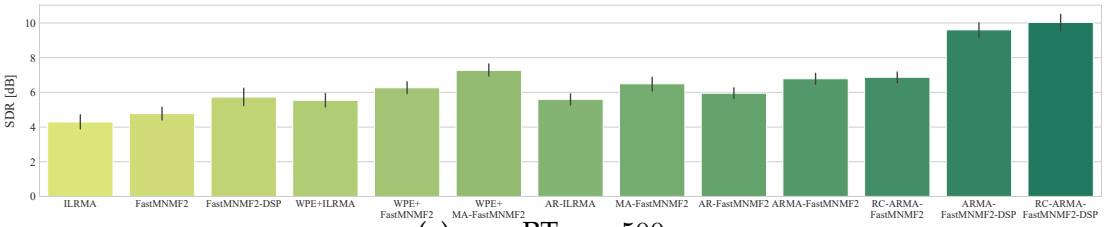
(b) far $\mathrm{RT}_{60} = 500\mathrm{ms}$



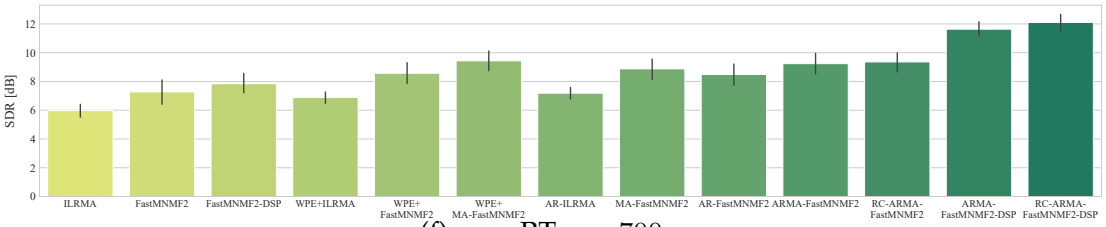
(c) far $\mathrm{RT}_{60} = 700\mathrm{ms}$



(d) near $\mathrm{RT}_{60} = 250\mathrm{ms}$

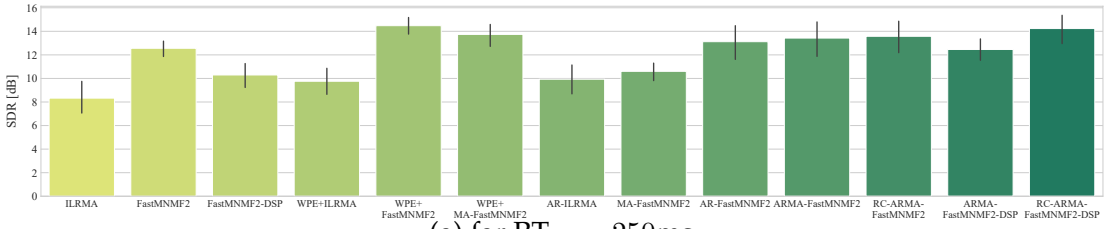


(e) near $\mathrm{RT}_{60} = 500\mathrm{ms}$

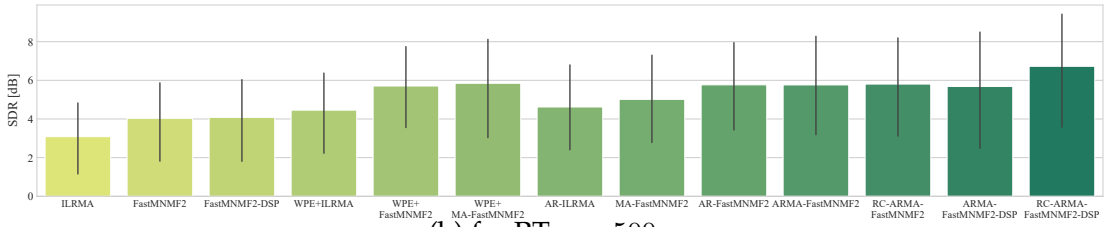


(f) near $\mathrm{RT}_{60} = 700\mathrm{ms}$

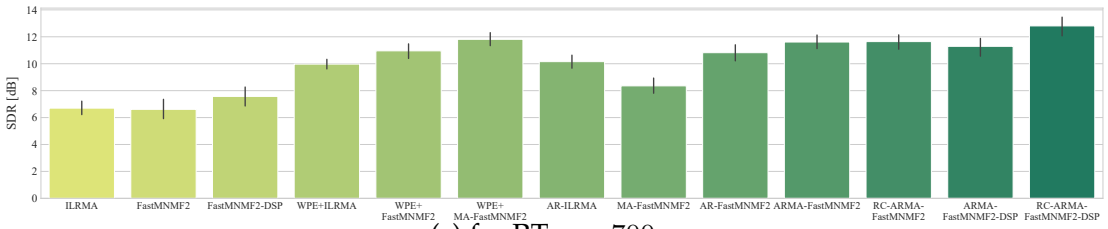
Figure 5.9: The average SDRs in speech enhancement ($M = 3$).



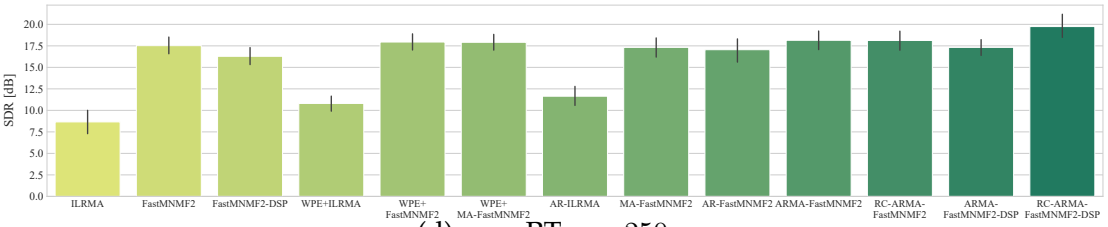
(a) far $\mathrm{RT}_{60} = 250\mathrm{ms}$



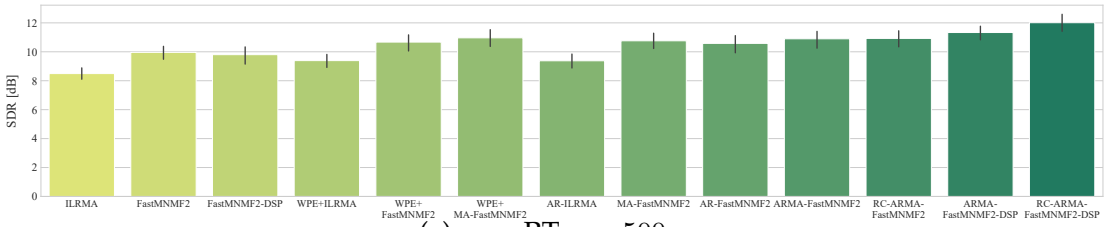
(b) far $\mathrm{RT}_{60} = 500\mathrm{ms}$



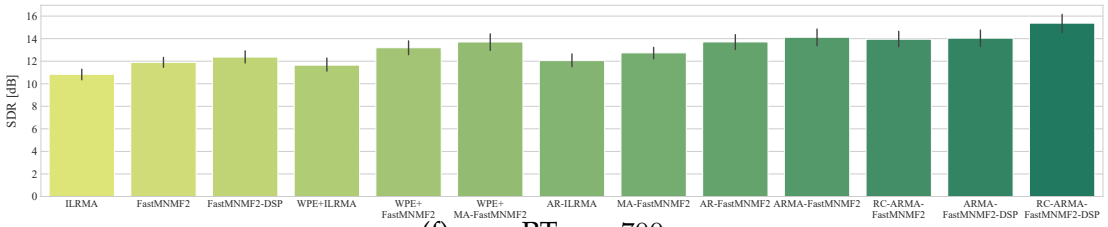
(c) far $\mathrm{RT}_{60} = 700\mathrm{ms}$



(d) near $\mathrm{RT}_{60} = 250\mathrm{ms}$



(e) near $\mathrm{RT}_{60} = 500\mathrm{ms}$



(f) near $\mathrm{RT}_{60} = 700\mathrm{ms}$

Figure 5.10: The average SDRs in speech enhancement ($M = 8$).

5.6 Summary

In this chapter, we proposed a joint source separation and dereverberation method called ARMA-FastMNMF2, which is an extension of FastMNMF2 based on the AR and MA models. The MA model represents the early reflection of each source, where the SCMs corresponding to the direct and reflection paths are restricted to jointly-diagonalizable matrices to reduce the computational cost, as in FastMNMF2, The AR model represents the late reverberation of the mixture as the weighted sum of the previous time frames. The long reverberation can be represented with a small tap length.

In the experiment using reverberant mixtures of two speech sources, AR-FastMNMF2 with $M = 3, 8$ worked well, but the performance was degraded when a longer tap length was used in a less-reverberant environment. Although the performance of MA-FastMNMF2 with $M = 3$ was limited because of the joint diagonalization constraint in the MA model, its computational cost is smaller than that of AR-FastMNMF2, and MA-FastMNMF2 with $M = 8$ and a long tap length worked well regardless of the reverberation time. We thus conclude that ARMA-FastMNMF2 with a short tap length of the AR model and a long tap length of the MA model worked robustly in many situations. In the experiment using noisy reverberant mixtures, ARMA-FastMNMF2 outperformed the conventional BSS methods and the sequential methods that use WPE and BSS methods sequentially. In the speech enhancement task, ARMA-FastMNMF2-DSP did not work well when the reverberation time was short because not only the reverberation but also the direct signal were represented by the MA model and suppressed. The rank constraint on the MA model alleviated this problem, and RC-ARMA-FastMNMF2-DSP outperformed ARMA-FastMNMF2 and other comparative methods in all cases.

Chapter 6

Conclusion

This thesis has addressed multichannel speech enhancement, source separation, and dereverberation. This chapter reviews the contributions of this thesis and addresses the future directions.

6.1 Contributions

In Chapter 3, we presented a new statistical framework that integrates a physically-founded linear model (multichannel spatial model) with a powerful deep speech generative model (single-channel speech model) in a principled manner. As a spatial model, we tested full-rank and rank-1 models. Note that MNMF [27] with richer expressive power often underperforms ILRMA [28] because MNMF is known to be sensitive to the initialization of SCMs and tends to get stuck at local optima. Interestingly, our full-rank model outperforms the rank-1 version even when the SCMs are initialized randomly. This indicates that the precise source model helps the estimation of SCMs and alleviates the initial value sensitivity.

One drawbacks of the full-rank spatial model is the heavy computational cost due to the high degree of freedom. In Chapter 4, we presented a jointly-diagonalizable (JD) full-rank spatial model and its application to MNMF called FastMNMF1 to reduce the computational cost of the full-rank spatial model. Then, taking the interpretation of the JD full-rank spatial model into account, we presented a well-behaved constrained version of FastMNMF1 called FastMNMF2 that shares the directional feature of each source over all frequency bins, and

rank-constrained version of FastMNMF1 and FastMNMF2. In the experiments, we showed that FastMNMF2 almost always outperformed FastMNMF1 and other conventional methods.

The recorded signals in real applications include reverberations in addition to environmental noise and non-target signals, while reverberations were not taken into account in the previous chapters. We presented joint source separation and dereverberation methods called MA-, AR-, and ARMA-FastMNMF2 in Chapter 5. These methods are extensions of FastMNMF2 that integrate either or both of an autoregressive (AR) process and moving-average (MA) process. In the experiment, MA-FastMNMF2 with a larger M worked well efficiently. AR-FastMNMF2 achieved better performance than MA-FastMNMF2 in many cases, with higher computational cost. ARMA-FastMNMF2 with a small tap length of the AR model and a long tap length of the MA model worked as well as AR-FastMNMF2 with a longer tap length with lower computational cost. Thus, ARMA-FastMNMF2 can be said to be robust and computationally-efficient blind method for joint source separation and dereverberation.

6.2 Future Work

This section describes several open problems regarding the methods developed in this thesis and future research directions.

- All the proposed methods assume that the number of sound sources is known in advance. One promising approach to estimating the number of sources or speakers at run-time is audio-visual integration based on object detection and lip reading. One may introduce time-varying latent variables that indicate whether each source is active or not, and put sparse priors on the variables for Bayesian inference.
- Another assumption is that the length of the observed signals is limited, and the number of sound sources and their positions are unchanged. For real-time applications, it is thus important to develop an online extension that can sequentially process the observed noisy reverberant mixtures with

small latency in a memory-efficient manner. The naive online extension based on mini-batches would suffer from the permutation problem because the speakers or their positions might be different between mini-batches. It is thus necessary to investigate an effective way of (partially) passing and adapting the current estimate of parameters to the next mini-batch.

- In this thesis, the powerful representation capability of deep learning was leveraged for improving only the source model, resulting in the deep speech prior. Recently, under an determined condition, independent vector analysis based on the normalizing flow (NF) called NF-IVA [93] was proposed for formulating a DNN-based spatial model with time-varying linear demixing matrices. Combining these approaches together, it would be possible to formulate a DNN-based integrated generative model whose parameters can be optimized jointly with gradient descent. While the spatial and source models have been considered separately in this thesis, it is worth investigating a way of directly formulating a unified model without clear distinction of the source and spatial models.

Bibliography

[1] D. V. Compernolle, W. Ma, F. Xie, and M. V. Diest, "Speech recognition in noisy environments with the aid of microphone arrays," *Speech Communication*, vol. 9, no. 5, pp. 433–442, 1990.

[2] M. Omologo, P. Svaizer, and M. Matassoni, "Environmental conditions and acoustic transduction in hands-free speech recognition," *Speech Communication*, vol. 25, no. 1, pp. 75–95, 1998.

[3] B. Li, T. N. Sainath, A. Narayanan, J. Caroselli, M. Bacchiani, A. Misra, I. Shafran, H. Sak, G. Punduk, K. Chin, K. C. Sim, R. J. Weiss, K. W. Wilson, E. Variani, C. Kim, O. Siohan, M. Weintraub, E. Mcdermott, R. Rose, and M. Shannon, "Acoustic modeling for google home," in *Interspeech*, pp. 399–403, 2017.

[4] Audio Software Engineering and Siri Speech Team, "Optimizing Siri on homepod in far-field settings," 2018. Apple Inc. [Online]. Available: https://machinelearning.apple.com/research/optimizing-siri-on-homepod-in-far-field-settings.

[5] R. Haeb-Umbach, S. Watanabe, T. Nakatani, M. Bacchiani, B. Hoffmeister, M. L. Seltzer, H. Zen, M. Souden, and Speech, "Speech processing for digital home assistants," *IEEE Signal Processing Magazine*, vol. 36, no. 6, pp. 111–124, 2019.

[6] S. Boll, "Suppression of acoustic noise in speech using spectral subtraction," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 27, no. 2, pp. 113–120, 1979.

[7] R. Martin, "Suppression of acoustic noise in speech using spectral subtraction," *IEEE Transactions on Speech Audio Processing*, vol. 9, no. 5, pp. 504–512,

2001.

[8] C. Févotte, N. Bertin, and J.-L. Durrieu, "Nonnegative matrix factorization with the Itakura-Saito divergence. with application to music analysis," *Neural Computation*, vol. 21, no. 3, pp. 793–830, 2009.

[9] X. Lu, Y. Tsao, S. Matsuda, and C. Hori, "Speech enhancement based on deep denoising autoencoder," in *Interspeech*, pp. 436–440, 2013.

[10] A. Narayanan and D. Wang, "Ideal ratio mask estimation using deep neural networks for robust speech recognition," in *International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, pp. 7092–7096, 2013.

[11] J. Heymann, L. Drude, and R. Haeb-Umbach, "Neural network based spectral mask estimation for acoustic beamforming," in *International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, pp. 196–200, 2016.

[12] H. Erdogan, J. Hershey, S. Watanabe, M. Mandel, and J. L. Roux, "Improved MVDR beamforming using single-channel mask prediction networks," in *Interspeech*, pp. 1981–1985, 2016.

[13] X. Li, J. Li, and Y. Yan, "Ideal ratio mask estimation using deep neural networks for monaural speech segregation in noisy reverberant conditions," in *Interspeech*, pp. 1203–1207, 2017.

[14] E. Vincent, T. Virtanen, and S. Gannot, eds., *Audio Source Separation and Speech Enhancement*. Wiley, 2018.

[15] K. Lebart, J. M. Boucher, and P. N. Denbigh, "A new method based on spectral subtraction for speech dereverberation," *Acta Acustica United with Acustica*, vol. 87, pp. 359–366, 2001.

[16] H. W. Löllmann and P. Vary, "Low delay noise reduction and dereverberation for hearing aids," *EURASIP Journal Advances in Signal Processing*, vol. 2009, 2009.

[17] L. Wang, K. Odani, and A. Kai, "Dereverberation and denoising based on generalized spectral subtraction by multi-channel lms algorithm using a small-scale microphone array," *EURASIP Journal on Advances in Signal Processing*, vol. 2012, 2012.

[18] K. Kinoshita, M. Delcroix, T. Nakatani, and M. Miyoshi, "Suppression

of late reverberation effect on speech signal using long-term multiple-step linear prediction," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 17, no. 4, pp. 534–545, 2009.

[19] T. Nakatani, T. Yoshioka, K. Kinoshita, M. Miyoshi, and Biing-Hwang Juang, "Speech dereverberation based on variance-normalized delayed linear prediction," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 18, no. 7, pp. 1717–1731, 2010.

[20] M. Togami and T. Komatsu, "Fast convergence algorithm for state-space model based speech dereverberation by multi-channel non-negative matrix factorization," in *IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA)*, pp. 239–243, 2019.

[21] B. D. V. Veen and K. M. Buckley, "Beamforming: A versatile approach to spatial filtering," *IEEE ASSP Magazine*, vol. 5, no. 2, pp. 4–24, 1988.

[22] A. Hyvärinen, J. Karhunen, and E. Oja, *Independent Component Analysis*. John Wiley & Sons, 2004.

[23] T. Kim, T. Eltoft, and T.-W. Lee, "Independent vector analysis: An extension of ICA to multivariate components," in *ICA*, pp. 165–172, 2006.

[24] N. Q. K. Duong, E. Vincent, and R. Gribonval, "Under-determined reverberant audio source separation using a full-rank spatial covariance model," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 18, no. 7, pp. 1830–1840, 2010.

[25] A. Ozerov and C. Févotte, "Multichannel nonnegative matrix factorization in convolutive mixtures for audio source separation," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 18, no. 3, pp. 550–563, 2010.

[26] S. Arberet, A. Ozerov, N. Duong, E. Vincent, R. Gribonval, F. Bimbot, and P. Vandergheynst, "Nonnegative matrix factorization and spatial covariance model for under-determined reverberant audio source separation," in *ISSPA*, pp. 1–4, 2010.

[27] H. Sawada, H. Kameoka, S. Araki, and N. Ueda, "Multichannel extensions of non-negative matrix factorization with complex-valued data," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 21, no. 5, pp. 971–

982, 2013.

[28] D. Kitamura, N. Ono, H. Sawada, H. Kameoka, and H. Saruwatari, "Determined blind source separation unifying independent vector analysis and nonnegative matrix factorization," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 24, no. 9, pp. 1626–1641, 2016.

[29] S. Makino, ed., *Audio Source Separation*. Springer, 2018.

[30] J. Azcarreta, N. Ito, S. Araki, and T. Nakatani, "Permutation-free cGMM: Complex gaussian mixture model with inverse wishart mixture model based spatial prior for permutation-free source separation and source counting," in *International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, pp. 51–55, 2018.

[31] H. Erdogan, J. R. Hershey, S. Watanabe, and J. Le Roux, "Phase-sensitive and recognition-boosted speech separation using deep recurrent neural networks," in *International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, pp. 708–712, 2015.

[32] K. Kinoshita, M. Delcroix, H. Kwon, T. Mori, and T. Nakatani, "Neural network-based spectrum estimation for online WPE dereverberation," in *Interspeech*, pp. 384–388, 2017.

[33] K. Shimada, Y. Bando, M. Mimura, K. Itoyama, K. Yoshii, and T. Kawahara, "Unsupervised speech enhancement based on multichannel NMF-informed beamforming for noise-robust automatic speech recognition," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 27, no. 5, pp. 960–971, 2019.

[34] A. Pandey and D. Wang, "On cross-corpus generalization of deep learning based speech enhancement," *arXiv preprint arXiv: 2002.04027*, 2020.

[35] N. Ito, S. Araki, and T. Nakatani, "FastFCA: A joint diagonalization based fast algorithm for audio source separation using a full-rank spatial covariance model," in *European Signal Processing Conference (EUSIPCO)*, pp. 1667–1671, 2018.

[36] N. Ito, C. Schymura, S. Araki, and T. Nakatani, "Noisy cGMM: Complex gaussian mixture model with non-sparse noise model for joint source sepa-

ration and denoising," in *European Signal Processing Conference (EUSIPCO)*, pp. 1676–1680, 2018.

[37] K. Sekiguchi, Y. Bando, K. Yoshii, and T. Kawahara, "Bayesian multichannel speech enhancement with a deep speech prior," in *APSIPA*, pp. 1233–1239, 2018.

[38] D. D. Lee and S. H. Seung, "Learning the parts of objects by non-negative matrix factorization," *Nature*, vol. 401, no. 6755, pp. 788–791, 1999.

[39] N. Murata, S. Ikeda, and A. Ziehe, "An approach to blind source separation based on temporal structure of speech signals," *Neurocomputing*, vol. 41, no. 1-4, pp. 1–24, 2001.

[40] Y. Bando, M. Mimura, K. Itoyama, K. Yoshii, and T. Kawahara, "Statistical speech enhancement based on probabilistic integration of variational autoencoder and non-negative matrix factorization," in *International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, pp. 716–720, 2018.

[41] S. Leglaive, L. Girin, and R. Horaud, "A variance modeling framework based on variational autoencoders for speech enhancement," in *IEEE MLSP*, pp. 1–6, 2018.

[42] K. Sekiguchi, Y. Bando, A. A. Nugraha, K. Yoshii, and T. Kawahara, "Semi-supervised multichannel speech enhancement with a deep speech prior," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 27, no. 12, pp. 2197–2212, 2019.

[43] K. Sekiguchi, Y. Bando, A. A. Nugraha, K. Yoshii, and T. Kawahara, "Fast multichannel nonnegative matrix factorization with directivity-aware jointly-diagonalizable spatial covariance matrices for blind source separation," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 28, pp. 2610–2625, 2020.

[44] N. Ito and T. Nakatani, "FastMNMF: Joint diagonalization based accelerated algorithms for multichannel nonnegative matrix factorization," in *International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, pp. 371–375, 2019.

[45] S. Seki, H. Kameoka, L. Li, T. Toda, and K. Takeda, "Generalized multichannel variational autoencoder for underdetermined source separation," in *European Signal Processing Conference (EUSIPCO)*, 2019.

[46] M. Togami and Y. Kawaguchi, "Noise robust speech dereverberation with kalman smoother," in *International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, pp. 7447–7451, 2013.

[47] M. Nakano, H. Kameoka, J. Le Roux, Y. Kitano, N. Ono, and S. Sagayama, "Convergence-guaranteed multiplicative algorithms for nonnegative matrix factorization with β-divergence," in *IEEE MLSP*, pp. 283–288, 2010.

[48] C. Févotte and A. T. Cemgil, "Nonnegative matrix factorizations as probabilistic inference in composite models," in *European Signal Processing Conference (EUSIPCO)*, pp. 1913–1917, 2009.

[49] K. Yoshii, "Correlated tensor factorization for audio source separation," in *International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, pp. 731–735, 2018.

[50] K. Yoshii, R. Tomioka, D. Mochihashi, and M. Goto, "Infinite positive semidefinite tensor factorization for source separation of mixture signals," in *ICML*, pp. 576–584, 2013.

[51] K. Yoshii, K. Kitamura, Y. Bando, E. Nakamura, and T. Kawahara, "Independent low-rank tensor analysis for audio source separation," in *European Signal Processing Conference (EUSIPCO)*, pp. 1671–1675, 2018.

[52] F. Weninger, J. R. Hershey, J. Le Roux, and B. Schuller, "Discriminatively trained recurrent neural networks for single-channel speech separation," in *GlobalSIP*, pp. 740–744, 2014.

[53] P.-S. Huang, M. Kim, M. Hasegawa-Johnson, and P. Smaragdis, "Deep learning for monaural speech separation," in *International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, pp. 1581–1585, 2014.

[54] J. R. Hershey, Z. Chen, J. Roux, and S. Watanabe, "Deep clustering : Discriminative embeddings for segmentation and separation," in *International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, pp. 31–35, 2016.

[55] Y. Isik, J. L. Roux, Z. Chen, S. Watanabe, and J. R. Hershey, "Single-channel multi-speaker separation using deep clustering," in *Interspeech*, pp. 545–549, 2016.

[56] D. Yu, M. Kolbaek, Z. H. Tan, and J. Jensen, "Permutation invariant training of deep models for speaker-independent multi-talker speech separation," in *International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, pp. 241–245, 2017.

[57] M. Kolbaek, D. Yu, Z.-H. Tan, and J. Jensen, "Multi-talker speech separation with utterance-level permutation invariant training of deep recurrent neural networks," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 25, no. 10, pp. 1901—-1913, 2017.

[58] Z. Chen, Y. Luo, and N. Mesgarani, "Deep attractor network for single-microphone speaker separation," in *International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, pp. 246–250, 2017.

[59] D. Michelsanti and Z.-H. Tan, "Conditional generative adversarial networks for speech enhancement and noise-robust speaker verification," in *Interspeech*, pp. 2008–2012, 2017.

[60] C. Donahue, B. Li, and R. Prabhavalkar, "Exploring speech enhancement with generative adversarial networks for robust speech recognition," in *International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, pp. 5024–5028, 2018.

[61] S. Pascual, A. Bonafonte, and J. Serra, "SEGAN: Speech enhancement generative adversarial network," in *Interspeech*, pp. 3642–3646, 2017.

[62] Y. Luo and N. Mesgarani, "TasNet: Surpassing ideal time-frequency masking for speech separation," in *International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, 2018.

[63] Y. Luo and N. Mesgarani, "Conv-TasNet: Surpassing ideal time-frequency magnitude masking for speech separation," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 27, no. 8, pp. 1256–1266, 2019.

[64] Y. Li and D. Wang, "On the optimality of ideal binary time– frequency masks," in *International Conference on Acoustics, Speech, and Signal Processing*

(ICASSP), pp. 3501–3504, 2009.

[65] J. Le Roux, S. Wisdom, H. Erdogan, and J. R. Hershey, "Sdr-half-baked or well done?," *arXiv preprint arXiv:1811.02508*, 2018.

[66] D. Wang and J. Chen, "Supervised speech separation based on deep learning: An overview," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 26, no. 10, 2018.

[67] P. Comon, "Independent component analysis, a new concept?," *Signal Processing*, pp. 287–314, 1994.

[68] P. Smaragdis, "Blind separation of convolved mixtures in the frequency domain," *Neurocomputing*, pp. 21–34, 1998.

[69] A. Hiroe, "Solution of permutation problem in frequency domain ICA using multivariate probability density functions," in *International Conference on Independent Component Analysis and Blind Signal Separation (ICA)*, pp. 601–608, 2006.

[70] N. Ono, "Stable and fast update rules for independent vector analysis based on auxiliary function technique," in *IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA)*, pp. 189–192, 2011.

[71] R. Scheibler and N. Ono, "Fast and stable blind source separation with rank-1 updates," in *International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, pp. 236–240, 2020.

[72] R. Scheibler, "Independent vector analysis via log-quadratically penalized quadratic minimization," *arXiv preprint arXiv: 2008.10048v1*, 2020.

[73] R. Scheibler and N. Ono, "MM algorithms for joint independent subspace analysis with application to blind single and multi-source extraction," *arXiv preprint arXiv: 2004.03926v1*, 2020.

[74] R. Scheibler and N. Ono, "Independent vector analysis with more microphones than sources," in *IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA)*, pp. 180–184, 2019.

[75] R. Ikeshita, T. Nakatani, and S. Araki, "Overdetermined independent vector analysis," in *International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, 2020.

[76] A. A. Nugraha, K. Sekiguchi, M. Fontaine, Y. Bando, and K. Yoshii, "Flow-based independent vector analysis for blind source separation," *IEEE Signal Processing Letters*, vol. 27, pp. 2173–2177, 2020.

[77] R. Ikeshita, "Independent positive semidefinite tensor analysis in blind source separation," in *European Signal Processing Conference (EUSIPCO)*, pp. 1652–1656, 2018.

[78] D. Kitamura, N. Ono, H. Sawada, H. Kameoka, and H. Saruwatari, "Relaxation of rank-1 spatial constraint in overdetermined blind source separation," in *European Signal Processing Conference (EUSIPCO)*, pp. 1271–1275, 2015.

[79] Y. Kubo, N. Takamune, D. Kitamura, and H. Saruwatari, "Efficient full-rank spatial covariance estimation using independent low-rank matrix analysis for blind source separation," in *European Signal Processing Conference (EUSIPCO)*, pp. 1–5, 2019.

[80] N. Ito and T. Nakatani, "FastFCA-AS: Joint diagonalization based acceleration of full-rank spatial covariance analysis for separating any number of sources," in *International Workshop on Acoustic Signal Enhancement (IWAENC)*, pp. 151–155, 2018.

[81] J. Nikunen and T. Virtanen, "Direction of arrival based spatial covariance model for blind sound source separation," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 22, no. 3, pp. 727–739, 2014.

[82] J. Nikunen and T. Virtanen, "Multichannel audio separation by direction of arrival based spatial covariance model and non-negative matrix factorization," in *International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, pp. 6677–6681, 2014.

[83] K. Sekiguchi, A. A. Nugraha, Y. Bando, and K. Yoshii, "Fast multichannel source separation based on jointly diagonalizable spatial covariance matrices," in *European Signal Processing Conference (EUSIPCO)*, 2019.

[84] S. Lee, S. H. Park, and K. M. Sung, "Beamspace-domain multichannel nonnegative matrix factorization for audio source separation," *IEEE Signal Processing Letters*, vol. 19, no. 1, pp. 43–46, 2012.

[85] Y. Mitsufuji, S. Koyama, and H. Saruwatari, "Multichannel blind source

separation based on non-negative tensor factorization in wavenumber domain," in *International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, pp. 56–60, 2016.

[86] T. Taniguchi and T. Masuda, "Linear demixed domain multichannel non-negative matrix factorization for speech enhancement," in *International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, pp. 476–480, 2017.

[87] Y. Mitsufuji, S. Uhlich, N. Takamune, D. Kitamura, S. Koyama, and H. Saruwatari, "Multichannel non-negative matrix factorization using banded spatial covariance matrices in wavenumber domain," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 2019.

[88] T. Otsuka, K. Ishiguro, H. Sawada, and H. G. Okuno, "Bayesian nonparametrics for microphone array processing," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 22, no. 2, pp. 493–504, 2014.

[89] N. Ito, S. Araki, T. Yoshioka, and T. Nakatani, "Relaxed disjointness based clustering for joint blind source separation and dereverberation," in *International Workshop on Acoustic Signal Enhancement (IWAENC)*, pp. 268–272, 2014.

[90] K. Itakura, Y. Bando, E. Nakamura, K. Itoyama, K. Yoshii, and T. Kawahara, "Bayesian multichannel audio source separation based on integrated source and spatial models," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 26, no. 4, pp. 831–846, 2018.

[91] H. Kameoka, L. Li, S. Inoue, and S. Makino, "Supervised determined source separation with multichannel variational autoencoder," *Neuro Computation*, vol. 31, no. 9, pp. 1–24, 2019.

[92] S. Leglaive, L. Girin, and R. Horaud, "Semi-supervised multichannel speech enhancement with variational autoencoders and non-negative matrix factorization," in *International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, pp. 101–105, 2019.

[93] A. A. Nugraha, K. Sekiguchi, and K. Yoshii, "A flow-based deep latent variable model for speech spectrogram modeling and enhancement,"

IEEE/ACM Transactions on Audio, Speech, and Language Processing, vol. 28, pp. 1104 – 1117, 2020.

[94] Y. Du, K. Sekiguchi, Y. Bando, A. A. Nugraha, M. Fontaine, K. Yoshii, and T. Kawahara, "Semi-supervised multichannel speech separation based on a phone- and speaker-aware deep generative model of speech spectrograms," in *European Signal Processing Conference (EUSIPCO)*, 2020.

[95] S. Mogami, H. Sumino, D. Kitamura, N. Takamune, S. Takamichi, H. Saruwatari, and N. Ono, "Independent deeply learned matrix analysis for multichannel audio source separation," in *European Signal Processing Conference (EUSIPCO)*, pp. 1557–1561, 2018.

[96] A. A. Nugraha, A. Liutkus, and E. Vincent, "Multichannel audio source separation with deep neural networks," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 24, no. 9, pp. 1652–1664, 2016.

[97] M. Souden, J. Benesty, and S. Affes, "On optimal frequency-domain multichannel linear filtering for noise reduction," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 18, no. 2, 2010.

[98] E. Warsitz and R. Haeb-Umbach, "Blind acoustic beamforming based on generalized eigenvalue decomposition," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 15, no. 5, pp. 1529–1539, 2007.

[99] R. Schmidt, "Multiple emitter location and signal parameter estimation," *IEEE Transactions on Antennas and Propagation*, vol. 34, no. 3, pp. 276–280, 1986.

[100] X. Xiao, S. Watanabe, H. Erdogan, L. Lu, J. Hershey, M. L. Seltzer, G. Chen, Y. Zhang, M. Mandel, and D. Yu, "Deep beamforming networks for multichannel speech recognition," in *International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, pp. 5745–5749, 2016.

[101] Z. Meng, S. Watanabe, J. R. Hershey, and H. Erdogan, "Deep long short-term memory adaptive beamforming networks for multichannel robust speech recognition," in *International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, pp. 271–275, 2017.

[102] J. Heymann, L. Drude, C. Boeddeker, P. Hanebrink, and R. Haeb-Umbach,

"Beamnet: End-to-end training of a beamformer-supported multi-channel ASR system," in *International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, pp. 5325–5329, 2017.

[103] L. Drude, J. Heymann, and R. Haeb-Umbach, "Unsupervised training of neural mask-based beamforming," in *International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, pp. 81–85, 2019.

[104] Y. Bando, Y. Sasaki, and K. Yoshii, "Deep bayesian unsupervised source separation based on a complex gaussian mixture model," in *IEEE MLSP*, 2019.

[105] D. P. Kingma, D. J. Rezende, S. Mohamed, and M. Welling, "Semi-supervised learning with deep generative models," in *NIPS*, pp. 3581–3589, 2014.

[106] B. Schwartz, S. Gannot, and E. A. Habets, "Online speech dereverberation using Kalman filter and em algorithm," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 23, no. 2, pp. 394–406, 2015.

[107] N. D. Gaubitch, P. A. Naylor, and D. B. Ward, "On the use of linear prediction for dereverberation of speech," in *International Workshop on Acoustic Signal Enhancement (IWAENC)*, pp. 99–102, 2003.

[108] M. Delcroix, T. Hikichi, and M. Miyoshi, "Dereverberation of speech signals based on linear prediction," in *International Conference on Spoken Language Processing*, pp. 877–881, 2004.

[109] P. A. Naylor and N. D. Gaubitch, "Speech dereverberation," in *International Workshop on Acoustic Signal Enhancement (IWAENC)*, 2005.

[110] N. D. Gaubitch, D. B. Ward, and P. A. Naylor, "Statistical analysis of the autoregressive modeling of reverberant speech," *The Journal of the Acoustical Society of America*, vol. 120, no. 6, pp. 4031–4039, 2006.

[111] T. Yoshioka, T. Nakatani, T. Hikichi, and M. Miyoshi, "Maximum likelihood approach to speech enhancement for noisy reverberant signals," in *IEEE ICASSP*, pp. 4585–4588, 2008.

[112] K. Kinoshita, M. Delcroix, T. Nakatani, and M. Miyoshi, "Suppression of late reverberation effect on speech signal using long-term multiple-

step linear prediction," *IEEE Transactions on Audio, Speech and Language Processing*, vol. 17, no. 4, pp. 534–545, 2009.

[113] T. Yoshioka and T. Nakatani, "Generalization of multi-channel linear prediction methods for blind mimo impulse response shortening," *IEEE Transactions on Audio, Speech and Language Processing*, vol. 20, no. 10, pp. 2707–2720, 2012.

[114] T. Nakatani, B. H. Juang, T. Yoshioka, K. Kinoshita, and M. Miyoshi, "Importance of energy and spectral features in gaussian source model for speech dereverberation," in *IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA)*, pp. 299–302, 2007.

[115] J. S. Bradley, H. Sato, and M. Picard, "On the importance of early reflections for speech in rooms," *The Journal of the Acoustic Society ofAmerica*, vol. 113, no. 6, pp. 3233–3244, 2003.

[116] I. Arweilera and J. M. Buchholz, "The influence of spectral characteristics of early reflections on speech intelligibility," *The Journal of the Acoustical Society of America*, vol. 130, no. 2, pp. 996–1005, 2011.

[117] S. Braun and E. A. Habets, "Online dereverberation for dynamic scenarios using a Kalman filter with a autoregressive model," *IEEE Signal Processing Letters*, no. 12, pp. 1741–1745.

[118] H. Kagami, H. Kameoka, and M. Yukawa, "Joint separation and dereverberation of reverberant mixtures with determined multichannel non-negative matrix factorization," in *International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, pp. 31–35, 2018.

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

[120] M. Togami, "Multi-channel speech source separation and dereverberation with sequential integration of determined and underdetermined models," in *International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, pp. 231–235, 2020.

[121] X. Xiao, S. Zhao, D. Hoang, H. Nguyen, X. Zhong, D. L. Jones, E. S. Chng, and H. Li, "The NTU-ADSC systems for reverberation challenge 2014," in *REVERB Workshop*, 2014.

[122] K. Han, Y. Wang, D. Wang, W. S. Woods, I. Merks, and T. Zhang, "Learning spectral mapping for speech dereverberation and denoising," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 23, no. 6, pp. 982–992, 2015.

[123] O. Ernst, S. E. Chazan, S. Gannot, and J. Goldberger, "Speech dereverberation using fully convolutional networks," in *European Signal Processing Conference (EUSIPCO)*, pp. 390–394, 2018.

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

[125] J. Heymann, L. Drude, R. Haeb-Umbach, K. Kinoshita, and T. Nakatani, "Frame-online DNN-WPE dereverberation," in *International Workshop on Acoustic Signal Enhancement (IWAENC)*, pp. 466–470, 2018.

[126] A. S. Subramanian, X. Wang, S. Watanabe, T. Taniguchi, D. Tran, and Y. Fujita, "An investigation of end-to-end multichannel speech recognition for reverberant and mismatch conditions," *arXiv preprint arXiv:1904.09049*, 2019.

[127] L. Drude, C. Boeddeker, J. Heymann, R. Haeb-Umbach, K. Kinoshita, M. Delcroix, and T. Nakatani, "Integrating neural network based beamforming and weighted prediction error dereverberation," *Interspeech*, 2018.

[128] T. Nakatani, C. Boeddeker, K. Kinoshita, R. Ikeshita, M. Delcroix, and R. Haeb-Umbach, "Jointly optimal denoising, dereverberation, and source separation," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 28, pp. 2267–2282, 2020.

[129] D. Ulyanov, A. Vedaldi, and V. Lempitsky, "Deep image prior," in *IEEE International Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018.

[130] C. Robert and G. Casella, *Monte Carlo Statistical Methods*. Springer, 2005.

[131] D. E. Rumelhart, G. E. Hinton, and R. J. Williams, "Learning representations by back-propagating errors," *Nature*, vol. 323, no. 6088, pp. 533–536, 1986.

[132] T. Ando, C.-K. Li, and R. Mathias, "Geometric means," *Linear Algebra and its Applications*, vol. 385, pp. 305–334, 2004.

[133] W.-H. Chen, "A review of geometric mean of positive definite matrices," *British Journal of Mathematics & Computer Science*, vol. 5, no. 1, pp. 1–12, 2015.

[134] D. P. Kingma and M. Welling, "Auto-encoding variational Bayes," in *ICLR*, 2014.

[135] J. Barker, R. Marxer, E. Vincent, and S. Watanabe, "The third 'CHiME' speech separation and recognition challenge: Analysis and outcomes," *Computer Speech & Language*, vol. 46, pp. 605–626, 2017.

[136] E. Vincent, R. Gribonval, and C. Févotte, "Performance measurement in blind audio source separation," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 14, no. 4, pp. 1462–1469, 2006.

[137] C. Raffel, B. McFee, E. J. Humphrey, J. Salamon, O. Nieto, D. Liang, and D. P. W. Ellis, "mir_eval: A transparent implementation of common MIR metrics," in *International Society for Music Information Retrieval Conference (ISMIR)*, pp. 367–372, 2014.

[138] A. W. Rix, J. G. Beerends, M. P. Hollier, and A. P. Hekstra, "Perceptual evaluation of speech quality (PESQ)-a new method for speech quality assessment of telephone networks and codecs," in *International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, pp. 749–752, 2001.

[139] C. H. Taal, R. C. Hendriks, R. Heusdens, and J. Jensen, "An algorithm for intelligibility prediction of time-frequency weighted noisy speech," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 19, no. 7, pp. 2125–2136, 2011.

[140] J. Garofalo, D. Graff, D. Paul, and D. Pallett, "CSR-I (WSJ0) complete," *Linguistic Data Consortium, Philadelphia*, 2007.

[141] D. P. Kingma and J. Lei Ba, "Adam: A method for stochastic optimization," in *ICLR*, 2015.

[142] X. Anguera, C. Wooters, and J. Hernando, "Acoustic beamforming for speaker diarization of meetings," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 15, no. 7, pp. 2011–2022, 2007.

[143] M. Aharon, M. Elad, and A. Bruckstein, "K-SVD: An algorithm for designing overcomplete dictionaries for sparse representation," *IEEE Trans. Signal Processing*, vol. 54, no. 11, pp. 4311–4322, 2006.

[144] Z.-Q. Wang, J. L. Roux, and J. R. Hershey, "Multi-channel deep clustering: Discriminative spectral and spatial embeddings for speaker-independent speech separation," in *International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, pp. 1–5, 2018.

[145] S. Inoue, H. Kameoka, L. Li, S. Seki, and S. Makino, "Joint separation and dereverberation of reverberant mixtures with multichannel variational autoencoder," in *International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, pp. 96–100, 2019.

[146] J. J. Carabias-Orti, J. Nikunen, T. Virtanen, and P. Vera-Candeas, "Multichannel blind sound source separation using spatial covariance model with level and time differences and nonnegative matrix factorization," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 26, no. 9, pp. 1512–1527, 2018.

[147] R. Scheibler, E. Bezzam, and I. Dokmanić, "Pyroomacoustics: A python package for audio room simulation and array processing algorithms," in *International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, pp. 351–355, 2018.

[148] R. Ikeshita, N. Ito, T. Nakatani, and H. Sawada, "A unifying framework for blind source separation based on a joint diagonalizability constraint," in *European Signal Processing Conference (EUSIPCO)*, pp. 1–5, 2019.

[149] K. Kinoshita, M. Delcroix, T. Yoshioka, T. Nakatani, E. Habets, R. Haeb-Umbach, V. Leutnant, A. Sehr, W. Kellermann, R. Maas, S. Gannot, and B. Raj, "The REVERB challenge : A common evaluation framework for dereverberation and recognition of reverberant speech," in *IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA)*, 2013.

List of Publications

Refereed International Journal Papers

1) Kouhei Sekiguchi, Yoshiaki Bando, Aditya Arie Nugraha, Kazuyoshi Yoshii, Tatsuya Kawahara: Fast Multichannel Nonnegative Matrix Factorization with Directivity-Aware Jointly-Diagonalizable Spatial Covariance Matrices for Blind Source Separation, *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, Vol. 28, pp. 2610-2625, 2020.
2) Kouhei Sekiguchi, Yoshiaki Bando, Aditya Arie Nugraha, Kazuyoshi Yoshii, Tatsuya Kawahara: Semi-supervised Multichannel Speech Enhancement with a Deep Speech Prior, *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, Vol. 27, No. 12, pp. 2197-2212, 2019.
3) Kouhei Sekiguchi, Yoshiaki Bando, Katsutoshi Itoyama, Kazuyoshi Yoshii: Layout Optimization of Cooperative Distributed Microphone Arrays Based on Estimation of Source Separation Performance, *Journal of Robotics and Mechatronics*, Vol. 29, No. 1, 2017.

Refereed International Conference Papers

4) Kouhei Sekiguchi, Yoshiaki Bando, Aditya Arie Nugraha, Mathieu Fontaine, Kazuyoshi Yoshii: Autoregressive Fast Multichannel Nonnegative Matrix Factorization for Joint Blind Source Separation and Dereverberation, *IEEE International Conference on Acoustics, Speech and Signal Processing* (ICASSP), 2021.
5) Kouhei Sekiguchi, Aditya Arie Nugraha, Yoshiaki Bando, Kazuyoshi Yoshii: Fast Multichannel Source Separation Based on Jointly Diagonalizable Spatial

Covariance Matrices, *European Association for Signal Processing* (EUSIPCO), 2019.

6) Kouhei Sekiguchi, Yoshiaki Bando, Kazuyoshi Yoshii, Tatsuya Kawahara: Bayesian Multichannel Speech Enhancement with a Deep Speech Prior, *Asia-Pacific Signal and Information Processing Association* (APSIPA), 2018.

7) Kouhei Sekiguchi, Yoshiaki Bando, K. Nakamura, K. Nakadai, Katsutoshi Itoyama, Kazuyoshi Yoshii: Online Simultaneous Localization and Mapping of Multiple Sound Sources and Asynchronous Microphone Arrays, *IEEE/RSJ International Conference on Intelligent Robots and Systems* (IROS), pp. 1973-1979, 2016.

8) Kouhei Sekiguchi, Yoshiaki Bando, Katsutoshi Itoyama, Kazuyoshi Yoshii: Optimizing the Layout of Multiple Mobile Robots for Cooperative Sound Source Separation, *IEEE/RSJ International Conference on Intelligent Robots and Systems* (IROS), pp. 5548–5554, 2015.