# DNN-Based Frequency-Domain Permutation Solver for Multichannel Audio Source Separation

Fumiya Hasuike*, Daichi Kitamura*, and Rui Watanabe†
*National Institute of Technology, Kagawa College, Kagawa, Japan
†Japan Advanced Institute of Science and Technology, Ishikawa, Japan

***Abstract*—This paper focuses on frequency-domain blind source separation (BSS) for audio signals. This technique estimates frequency-wise source components from an observed spectrogram. Full-rank spatial covariance analysis and frequency-domain independent component analysis are algorithms commonly used for this task. Using these methods, however, results in an alignment problem of frequency-wise permutations of the estimated source components. This is known as the permutation problem, which has been addressed for decades and requires a robust and precise permutation solver post-processing. We introduce a permutation solver that uses a deep neural network and predicts the correct source permutations in each frequency. The experimental results demonstrate the validity of the proposed approach and its robustness against the domain of the dataset.**

## I. Introduction

Multichannel audio source separation is a technique for estimating source signals from an observed multichannel mixture signal. The methods that do not require a priori spatial information, e.g., locations of sources and microphones, are called blind source separation (BSS) [1]. For a determined situation (in which the numbers of microphones and sources are the same), independent component analysis (ICA) [2] and its variants have widely been utilized in BSS tasks.

A typical approach for audio BSS is frequency-domain ICA (FDICA) [3]. In FDICA, the observed multichannel signal is converted to the time-frequency domain by a short-time Fourier transform (STFT). Then ICA is independently applied to each frequency to estimate the separated source components. However, using FDICA leads to the permutation problem which will be addressed in this paper. Because ICA has permutation indeterminacy of estimated signals, the source order of ICA outputs depends on the initial values of parameters. For this reason, the estimated source components of FDICA are not aligned along the frequencies, as shown in Fig. 1. This is known as the permutation problem, which often arises even in underdetermined BSS, e.g., full-rank spatial covariance analysis [4].

Various permutation solvers with hand-crafted criteria have been studied in the context of FDICA [5], [6], [7], [8]. Subsequently, the mainstream approach for BSS shifted to developing a simultaneous solution of frequency-wise separation and the permutation alignment. In these methods, to avoid the permutation problem, the time-frequency structure of each source is assumed as a *source model*, which is then combined with FDICA. For example, independent vector analysis (IVA) [9], [10], [11], [12] assumes group sparsity in the spectrogram of

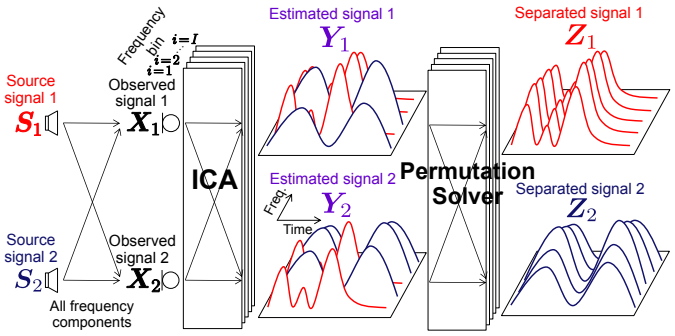


Fig. 1. Permutation problem in FDICA ($N = M = 2$).

each source as the source model. Independent low-rank matrix analysis (ILRMA) [13], [14] assumes that sources inherently have a low-rank time-frequency structure and models them by nonnegative matrix factorization [15]. Independent deeply learned matrix analysis [16] utilizes a deep neural network (DNN) that is trained by using specific audio datasets. The source model based on time-frequency masks has also been utilized [17], [18]. These methods provide more accurate BSS when the source model fits well with each of the sources in the observed signal. However, in [19], we found that FDICA with an ideal permutation solver (using oracle source signals) significantly outperforms IVA and ILRMA. This suggests that FDICA has the potential to achieve high-quality BSS, and the remaining task is to solve the permutation problem. The challenge is building a universal source model that fits various sources (speech signals, vocals, musical instruments, background noise, etc.).

To directly obtain such a versatile model, DNN-based supervised approaches with a large dataset and extensive training have shown potential for speech enhancement and audio source separation. Various methods have been proposed, such as those in Ref. [20], [21], [22]. In these techniques, the size and versatility of the training data are crucial for obtaining the universal source model and improving separation performance, but collecting and producing such training datasets is costly. Thus, BSS (without model training) or a separation technique with few-shot learning is still important.

We previously proposed an approach that utilizes a DNN to solve the permutation problem rather than building versatile source models [23]. This approach only requires a few-shot audio dataset because the permutation problem can easily be simulated by randomly shuffling the frequency components of the audio spectrogram. In the experimental section, we will

show that the permutation problem can be mostly solved by using training data that are produced from only two 11-s-long audio signals. This is an advantage of the permutation solver over various DNN-based supervised approaches.

In our previous method, the DNN was trained to predict whether the source components of the reference frequency and the other frequencies belong to the same source as a binary classification, where the source components are estimated by FDICA. Because predictions are performed for each subband, implementing the permutation solver based on the predicted results becomes a complicated process, particularly for three or more sources. In addition, the performance of the method depends on the type of sources used in the training dataset.

In this paper, to solve the above-mentioned problems, we propose a simple algorithm for the DNN-based permutation solver, which can easily extend to three or more sources. We demonstrate its performance and robustness against the dataset domain to evaluate its versatility.

## II. FDICA and Permutation Problem

### A. Definitions of Signals

Let the source and observed signals in the time-frequency domain be

$$\boldsymbol{s}_{ij} = [s_{ij1}, s_{ij2}, \cdots, s_{ijn}, \cdots, s_{ijN}]^{\mathrm{T}} \in \mathbb{C}^N, \quad (1)$$

$$\boldsymbol{x}_{ij} = [x_{ij1}, x_{ij2}, \cdots, x_{ijm}, \cdots, x_{ijM}]^{\mathrm{T}} \in \mathbb{C}^M, \quad (2)$$

respectively, where $i = 1, 2, \cdots, I$, $j = 1, 2, \cdots, J$, $n = 1, 2, \cdots, N$, and $m = 1, 2, \cdots, M$ are the indices of frequency, time, source, and channel, respectively. Also, let the estimated signal obtained by FDICA be

$$\boldsymbol{z}_{ij} = [z_{ij1}, z_{ij2}, \cdots, z_{ijn'}, \cdots, z_{ijN}]^{\mathrm{T}} \in \mathbb{C}^N, \quad (3)$$

where the indices $n$ and $n'$ are used properly to represent the ambiguity of a source permutation, e.g., the estimated signal of $s_{ij1}$ could be $z_{ij2}$. As we only focus on the determined situation, $N = M$ is assumed throughout the paper. In addition, we also define time-frequency matrices (complex-valued spectrograms) of the $n$th source, the $m$th observed, and the $n'$th estimated signals as $\boldsymbol{S}_n \in \mathbb{C}^{I\times J}$, $\boldsymbol{X}_m \in \mathbb{C}^{I\times J}$, and $\boldsymbol{Z}_{n'} \in \mathbb{C}^{I\times J}$, respectively.

### B. BSS Formulation and FDICA

FDICA [3] assumes the frequency-wise mixing system [19], [24]:

$$\boldsymbol{x}_{ij} = \boldsymbol{A}_i \boldsymbol{s}_{ij}, \quad (4)$$

where $\boldsymbol{A}_i \in \mathbb{C}^{M\times N}$ is a time-invariant frequency-wise mixing matrix. When the mixing matrix $\boldsymbol{A}_i$ is nonsingular, there exists a frequency-wise demixing matrix $\boldsymbol{W}_i = \boldsymbol{A}_i^{-1} \in \mathbb{C}^{N\times M}$. With this ideal demixing matrix, the estimated signal can be obtained as

$$\boldsymbol{z}_{ij} = \boldsymbol{W}_i \boldsymbol{x}_{ij}. \quad (5)$$

Thus, FDICA independently applies ICA to the complex-valued time-series signals of each frequency, $(\boldsymbol{x}_{ij})_{j=1}^J$, and estimates the frequency-wise demixing matrix $\boldsymbol{W}_i$ over all the frequencies.

### C. Permutation Problem

Because ICA has indeterminacy of scale and permutation of the estimated signals, the demixing matrix obtained by FDICA, $\hat{\boldsymbol{W}}_i \in \mathbb{C}^{N\times M}$, is represented as

$$\hat{\boldsymbol{W}}_i = \boldsymbol{D}_i \boldsymbol{P}_i \boldsymbol{W}_i, \quad (6)$$

where $\boldsymbol{D}_i \in \mathbb{R}^{N\times N}$ is a diagonal matrix that may change the scale of $(\boldsymbol{z}_{ij})_{j=1}^J$ and $\boldsymbol{P}_i \in \{0,1\}^{N\times N}$ is a permutation matrix that may replace the source order of $(\boldsymbol{z}_{ij})_{j=1}^J$. For example,

$$\boldsymbol{P}_i = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \text{ or } \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} \text{ (for } N = 2), \quad (7)$$

$$\boldsymbol{P}_i = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}, \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 1 & 0 \end{bmatrix}, \begin{bmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix}, \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \end{bmatrix}, \begin{bmatrix} 0 & 0 & 1 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix}, \text{ or } \begin{bmatrix} 0 & 0 & 1 \\ 0 & 1 & 0 \\ 1 & 0 & 0 \end{bmatrix} \text{ (for } N = 3). \quad (8)$$

The permutation matrix is a doubly stochastic matrix (DSM) [25], [26] because the summations of rows and columns are all unity.

As a result, the estimated signal of FDICA,

$$\boldsymbol{y}_{ij} = \hat{\boldsymbol{W}}_i \boldsymbol{x}_{ij} = [y_{ij1}, y_{ij2}, \cdots, y_{ijn'_i} \cdots, y_{ijN}]^{\mathrm{T}} \in \mathbb{C}^N, \quad (9)$$

includes inconsistent scales and permutations along the frequencies, where $n'_i = 1, 2, \cdots, N$ is a new source index to represent the permutation ambiguity in each frequency. The scale ambiguity caused by $\boldsymbol{D}_i$ can be easily recovered by applying a projection-back technique [27]. However, it is not easy to align the source permutations along the frequencies, which is the permutation problem. Fig. 1 shows a schematic of the problem, where $\boldsymbol{Y}_{n'} \in \mathbb{C}^{I\times J}$ is the $n'$th spectrogram of the estimated signal $\boldsymbol{y}_{ij}$ (including the permutation problem).

The ideal permutation solver is defined as $\boldsymbol{z}_{ij} = \boldsymbol{P}_i^{-1}\boldsymbol{D}_i^{-1}\boldsymbol{y}_{ij}$. Thus, solving this problem is interpreted as an estimation of $\boldsymbol{P}_i^{-1}$ over all frequencies. Strictly speaking, however, even if the source permutations are aligned correctly, the indeterminacy of permutation of entire frequency components (i.e., the sources) remains. Thus, the separated signal is represented as

$$\boldsymbol{z}_{ij} = \boldsymbol{P}_{\mathrm{all}} \boldsymbol{P}_i^{-1} \boldsymbol{D}_i^{-1} \boldsymbol{y}_{ij}, \quad (10)$$

where $\boldsymbol{P}_{\mathrm{all}} \in \{0,1\}^{N\times N}$ is a frequency-independent permutation matrix. The estimation of $\boldsymbol{P}_{\mathrm{all}}^{-1}$ is out of the scope of this paper.

## III. Proposed method

### A. Motivation

As described in Sect. I and [23], the DNN-based permutation solver directly assists FDICA-based BSS rather than the source model tailored to the specific source types. In addition, the

training data (signals with the permutation problem) can easily be produced by randomly shuffling the frequency components of source signals, which enables few-shot learning. For these reasons, this paper only focuses on developing a simple, robust, and precise DNN-based permutation solver.

The conventional DNN-based permutation solver [23] trains a DNN that solves the permutation problem within a specific subband and applies the DNN to all subbands. In each subband, the center frequency is defined as the reference frequency. The DNN predicts whether the components in the reference frequency and the other frequencies (within the subband) belong to the same source. When $N = 2$, this binary classification coincides with estimating a specific source permutation. However, when $N \geq 3$, we cannot determine the source permutation when the DNN outputs that the two components are not the same source. Therefore, we need to apply the DNN to all of the pair combinations from $N$ sources. Furthermore, the post-process for solving the permutation problem among subbands, which is a stitching technique along subbands, becomes more complicated as the number of sources increases. To solve this problem, we propose a simple DNN-based permutation solver that directly predicts the permutation matrix $\boldsymbol{P}_i^{-1}$ for all frequencies simultaneously.

### B. Input and Output of DNN

FDICA outputs spectrograms of the estimated signal, $(\boldsymbol{Y}_{n'})_{n'=1}^{N}$, including the permutation problem. As a preprocess, normalized power spectrograms are calculated as

$$\overline{\boldsymbol{Y}}_{n'} = \frac{|\boldsymbol{Y}_{n'}|^{.2}}{\sum_{n'=1}^{N} |\boldsymbol{Y}_{n'}|^{.2}} \in [0,1]^{I \times J}, \tag{11}$$

where $|\cdot|^{\cdot 2}$ for matrices denotes an element-wise absolute and square operation. This normalization stabilizes the training of the DNN and enhances correlations between the same source components [6]. Then, we extract a temporally local spectrogram from $(\overline{\boldsymbol{Y}}_{n'})_{n'=1}^{N}$ centered at time $j$ as

$$\check{\boldsymbol{Y}}_{jn'} = [\overline{\boldsymbol{y}}_{(j-\beta)n'} \ \overline{\boldsymbol{y}}_{(j-\beta+1)n'} \ \cdots \ \overline{\boldsymbol{y}}_{(j+\beta)n'}] \in [0,1]^{I \times (2\beta+1)}, \tag{12}$$

where $\overline{\boldsymbol{y}}_{jn'} \in [0,1]^I$ is the $j$th column vector of $\overline{\boldsymbol{Y}}_{n'}$. $\beta > 0$ is a hyperparameter that determines the duration of the local spectrogram $\check{\boldsymbol{Y}}_{jn'}$. The input of the proposed DNN is a flattened vector of $(\check{\boldsymbol{Y}}_{jn'})_{n'=1}^{N}$:

$$\boldsymbol{d}_j = \left[\mathrm{vec}(\check{\boldsymbol{Y}}_{j1})^{\mathrm{T}}, \cdots, \mathrm{vec}(\check{\boldsymbol{Y}}_{jN})^{\mathrm{T}}\right]^{\mathrm{T}} \in [0,1]^{NI(2\beta+1)}, \tag{13}$$

where $\mathrm{vec}(\cdot)$ denotes the vectorization of an input matrix.

As shown in Fig. 2, the DNN's prediction can be defined as

$$\hat{\boldsymbol{L}}_j = \mathrm{DNN}(\boldsymbol{d}_j) \in [0,1]^{I \times N!}, \tag{14}$$

where $\hat{\boldsymbol{L}}_j$ is a matrix that consists of the probabilities $\hat{l}_{iqj}$ for the permutation matrices, and $q = 1, 2, \cdots, N!$ is the index of the permutations. For example, when $N = 2$, the predicted

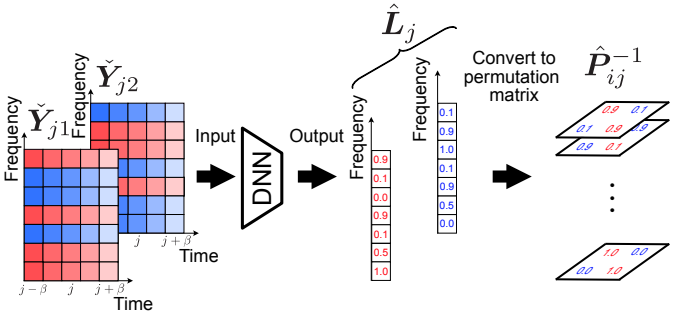


Fig. 2. Calculation of predicted permutation matrix for $N = 2$ case.

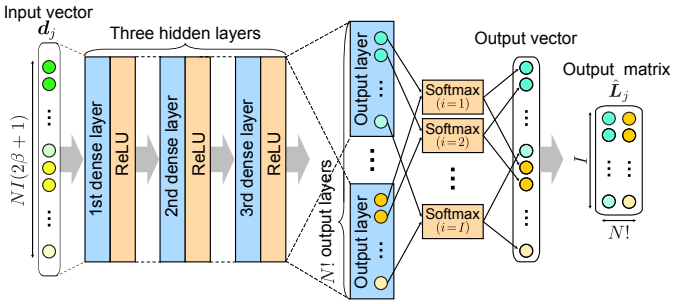


Fig. 3. DNN architecture for $N = 2$ case.

permutation matrix can be constructed by using $\hat{l}_{i1j}$ and $\hat{l}_{i2j}$ as

$$\hat{\boldsymbol{P}}_{ij}^{-1} = \hat{l}_{i1j} \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} + \hat{l}_{i2j} \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} \in [0,1]^{N \times N}. \tag{15}$$

When $N = 3$, the probabilities $(\hat{l}_{iqj})_{q=1}^{3!}$ are predicted, and $\hat{\boldsymbol{P}}_{ij}^{-1}$ is constructed as a linear combination of the matrices (8). Note that $\sum_q \hat{l}_{iqj} = 1$ is constrained in the DNN.

Thus, the proposed DNN directly predicts the correct permutation matrix for all frequencies simultaneously. From a different perspective, because (15) is a DSM, the DNN predicts coefficients of a convex combination of the true permutation matrices, which is known as the Birkhoff–von Neumann theorem [28].

### C. DNN Architecture

Fig. 3 shows the architecture of the DNN used in the proposed method. All of the layers consist of fully connected layers. Rectified linear unit (ReLU) functions [29] are used from the first to the third hidden layers. The output layer explicitly branches into $N!$ ($I$-dimensional) vectors with fully connected (non-shared) weights, and the frequency-wise softmax function is applied to each of the elements to ensure $\sum_q \hat{l}_{iqj} = 1$.

### D. Loss Function

The process after obtaining $\hat{\boldsymbol{P}}_{ij}^{-1}$ is shown in Fig. 4, where $(\check{\boldsymbol{Y}}_{jn'})_{n'=1}^{N}$ is the temporally local spectrograms of $(\boldsymbol{Y}_{n'})_{n'=1}^{N}$ (before the normalization (11)) extracted in the same manner as (12). First, $(\check{\boldsymbol{Y}}_{jn'})_{n'=1}^{N}$ is softly aligned by multiplying the predicted permutation matrices $(\hat{\boldsymbol{P}}_{ij}^{-1})_{i=1}^{I}$ in the same manner as (10), where the obtained spectrograms are defined

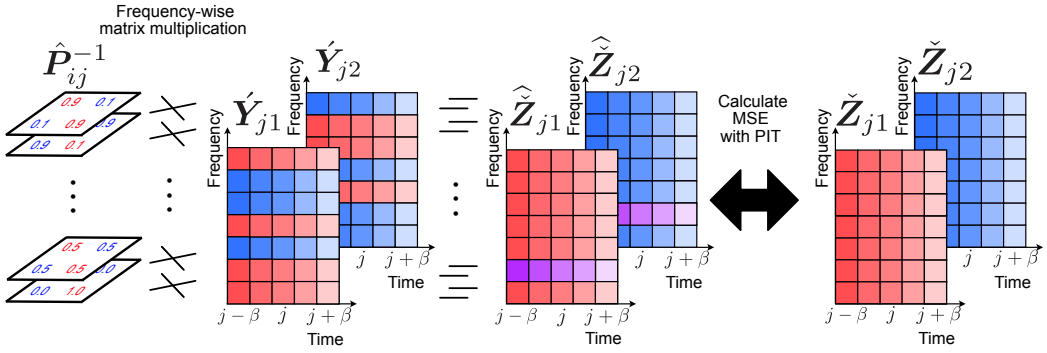


Fig. 4. Calculation of loss function using MSE with PIT for $N = 2$ case.

TABLE I
SPEECH AND MUSIC SOURCES OBTAINED FROM SISEC2011 [31]

| Signal type | Source | Data name | Length |
|---|---|---|---|
| Speech | Male speech | dev2_male4_inst_src_2.wav | 10.0 s |
| | Female speech | dev3_female4_inst_src_2.wav | 10.0 s |
| Music | Drums | dev1_wdrums_src_3.wav | 11.0 s |
| | Bass | dev1_wdrums_src_2.wav | 11.0 s |

as $(\hat{\check{\boldsymbol{Z}}}_{n'})_{n'=1}^{N}$. Then, we prepare the correctly aligned local spectrograms $(\check{\boldsymbol{Z}}_{n'})_{n'=1}^{N}$ for $(\boldsymbol{Y}_{n'})_{n'=1}^{N}$ as the label signals. Finally, the mean squared error (MSE) between $(\hat{\check{\boldsymbol{Z}}}_{n'})_{n'=1}^{N}$ and $(\check{\boldsymbol{Z}}_{n'})_{n'=1}^{N}$ is calculated as a loss function value of the DNN. As mentioned in Sect. II-C, we do not estimate the correct order of the sources, i.e., $\boldsymbol{P}_{\mathrm{all}}^{-1}$. To permit this source-order ambiguity while training the DNN, permutation invariant training (PIT) [30] is introduced. The loss function $\mathcal{L}$ is defined as

$$\mathcal{L} = \min(C_1, C_2, \cdots, C_{N!}), \tag{16}$$

$$C_q = \sum_{n'}^{N} ||\hat{\check{\boldsymbol{Z}}}_{jn'} - \check{\boldsymbol{Z}}_{j\mathcal{P}(q,n')}||_2^2, \tag{17}$$

where $\min(\cdot)$ is a minimum value of inputs and $\mathcal{P}(q, n')$ returns the $n'$th scalar in the $q$th permutation of all possible permutations.

### E. Application of Pretrained DNN to Test Data

When we apply the trained DNN to test data $(\boldsymbol{Y}_{n'})_{n'=1}^{N}$, the temporally local spectrograms $(\check{\boldsymbol{Y}}_{jn'})_{n'=1}^{N}$ with various time $j$ can be input to the DNN. Because the permutation matrix $\boldsymbol{P}_i$ is time-invariant, the effect of prediction errors can be mitigated by taking the majority decision among the predicted permutation matrices $(\hat{\boldsymbol{P}}_{ij}^{-1})_{j=1}^{J}$ as

$$\hat{\boldsymbol{P}}_i^{-1} = \mathrm{round}\left(\frac{1}{J}\sum_{j=1}^{J}\hat{\boldsymbol{P}}_{ij}^{-1}\right) \in \{0,1\}^{N\times N}, \tag{18}$$

where $\mathrm{round}(\cdot)$ denotes the element-wise rounding off. The separated signal is obtained by

$$\hat{\boldsymbol{z}}_{ij} = \hat{\boldsymbol{P}}_i^{-1}\boldsymbol{y}_{ij}. \tag{19}$$

## IV. EXPERIMENTS

### A. Conditions

We conducted experiments to compare performances of the conventional [23] and proposed methods. Furthermore, we show the robustness of the proposed method against the domain of the dataset, which demonstrates the validity of the DNN-based permutation solvers.

In this experiment, two pairs of dry source signals shown in Table I were used as $(\boldsymbol{S}_n)_{n=1}^{2}$. These sources were obtained from SiSEC2011 [31]. The sampling frequency of these signals was 16 kHz. STFT was performed using the 128-ms-long Hann window with 64-ms-long shifting, resulting in $I = 1025$ and $J = 158$ (speech signal) or $J = 173$ (music signal). To simulate the block permutation problem [32], which often arises in IVA and ILRMA, the entire frequencies of $(\boldsymbol{S}_n)_{n=1}^{2}$ were divided into 64 blocks (each block containing 16 frequency bins), and these blocks were randomly swapped between $\boldsymbol{S}_1$ and $\boldsymbol{S}_2$ to produce the input signal of the permutation solvers, i.e., $(\boldsymbol{Y}_{n'})_{n'=1}^{2}$. The training dataset for the conventional and proposed DNNs was generated from 300 random swapping patterns. Then, ten new patterns of randomly swapped signals were used as the test dataset for evaluation.

For the conventional method, the conditions of the DNN model, hyperparameters, and optimization were set to the same as those in [23]. For the proposed method, all of the hidden layers had 4096 units, and the number of local time frames was set to $\beta = 13$. We used the Adam optimizer with the minibatch size set to eight, and the number of training epochs was 1000. For both the conventional and proposed methods, we trained two DNN models, the speech and the music models, using only the speech and music signals, respectively. Then, we evaluated these models using the test dataset of both speech and music mixtures, i.e., *in-domain* (using the same sources in the training and testing) and *out-domain* (using different sources in the training and testing) evaluations. We used source-to-distortion ratio (SDR) [33] as the evaluation score, which represents accuracy and quality of BSS.

### B. Results

Figs. 5 and 6 show examples of permutation-simulated and aligned signals estimated by the conventional and proposed methods for the in-domain test dataset. Both the conventional

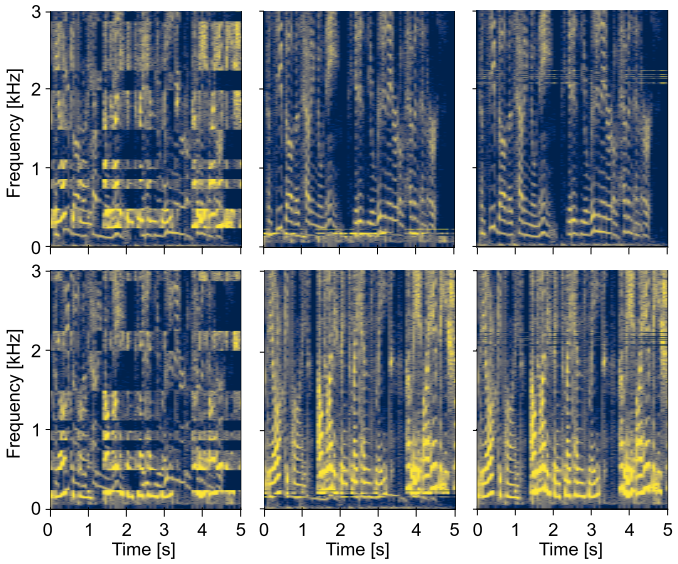


Fig. 5. Input (left) and estimated signals of conventional (center) and proposed methods (right): male (top) and female (bottom) speech.

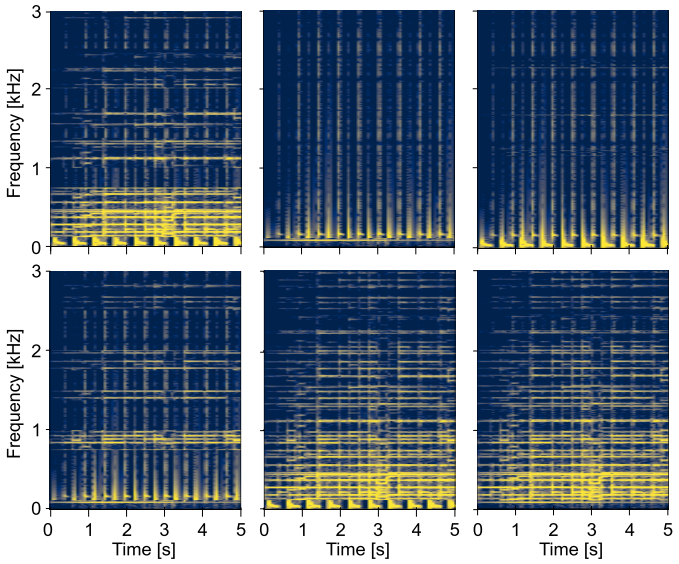


Fig. 6. Input (left) and estimated signals of conventional (center) and proposed methods (right): drums (top) and bass (bottom).

and proposed methods solve the permutation problem with high accuracy. However, the conventional method often fails to align the sources in a low-frequency band, which is crucial for SDR and subjective listening. Figs. 7 and 8 show the SDR values for the in-domain and out-domain test datasets, respectively. The values of the observed and estimated signals shown are averaged over the two sources. For both the speech and music results of the in-domain evaluation, the proposed method improved SDR by over 20 dB on average. In contrast, the conventional method often failed to improve SDR, particularly for the music signals. Furthermore, the efficacy of the proposed method can be verified even in the out-domain evaluation. This result shows the robustness against the domain of the dataset used to train the proposed DNN.

## V. Conclusion

We proposed a DNN-based permutation solver that directly predicts the permutation matrices for all frequencies simultaneously. The experimental results show significant improvement from the conventional method. We also verified the robustness against the domain of the dataset. The combination of a BSS technique and the proposed permutation solver for three or more sources will be compared with other conventional approaches in future work. In addition, explicit learning of dependencies along frequencies should be introduced.

## Acknowledgment

This work was partly supported by JSPS KAKENHI Grant Numbers 22H03652 and 19H01116. Also, the authors would like to thank Shuhei Yamaji for his support on the experiment.

## References

[1] H. Sawada, N. Ono, H. Kameoka, D. Kitamura, and H. Saruwatari, "A review of blind source separation methods: Two converging routes to ILRMA originating from ICA and NMF," *APSIPA Trans. Signal and Info. Process.*, vol. 8, no. e12, pp. 1–14, 2019.

[2] P. Comon, "Independent component analysis, a new concept?," *Signal Process.*, vol. 36, no. 3, pp. 287–314, 1994.

[3] P. Smaragdis, "Blind separation of convolved mixtures in the frequency domain," *Neurocomputing*, vol. 22, pp. 21–34, 1998.

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

[5] 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.

[6] H. Sawada, R. Mukai, S. Araki, and S. Makino, "A robust and precise method for solving the permutation problem of frequency-domain blind source separation," *IEEE Trans. Speech and Audio Process.*, vol. 12, no. 5, pp. 530–538, 2004.

[7] H. Saruwatari, T. Kawamura, T. Nishikawa, A. Lee, and K. Shikano, "Blind source separation based on a fast-convergence algorithm combining ICA and beamforming," *IEEE Trans. Audio, Speech, Lang. Process.*, vol. 14, no. 2, pp. 666–678, 2006.

[8] H. Sawada, S. Araki, and S. Makino, "Measuring dependence of bin-wise separated signals for permutation alignment in frequency-domain BSS," in *Proc. ISCAS*, pp. 3247–3250, 2007.

[9] A. Hiroe, "Solution of permutation problem in frequency domain ICA using multivariate probability density functions," in *Proc. ICA*, pp. 601–608, 2006.

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

[11] T. Kim, H. T. Attias, S.-Y. Lee, and T.-W. Lee, "Blind source separation exploiting higher-order frequency dependencies," *IEEE Trans. Audio, Speech, Lang. Process.*, vol. 15, no. 1, pp. 70–79, 2007.

[12] N. Ono, "Stable and fast update rules for independent vector analysis based on auxiliary function technique," in *Proc. WASPAA*, pp. 189–192, 2011.

[13] D. Kitamura, N. Ono, H. Sawada, H. Kameoka, and H. Saruwatari, "Determined blind source separation unifying independent vector analysis

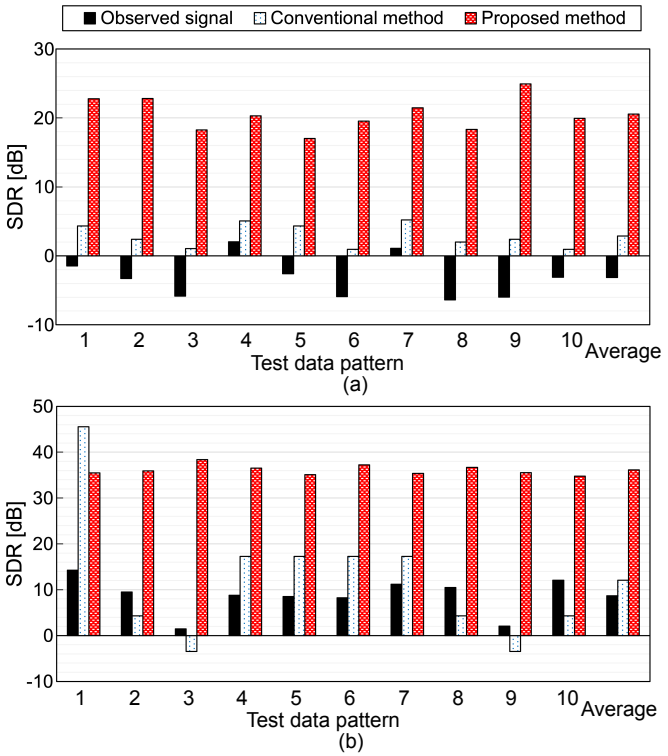


Fig. 7. Average SDR for in-domain test data: (a) speech mixtures with speech models and (b) music mixtures with music models.

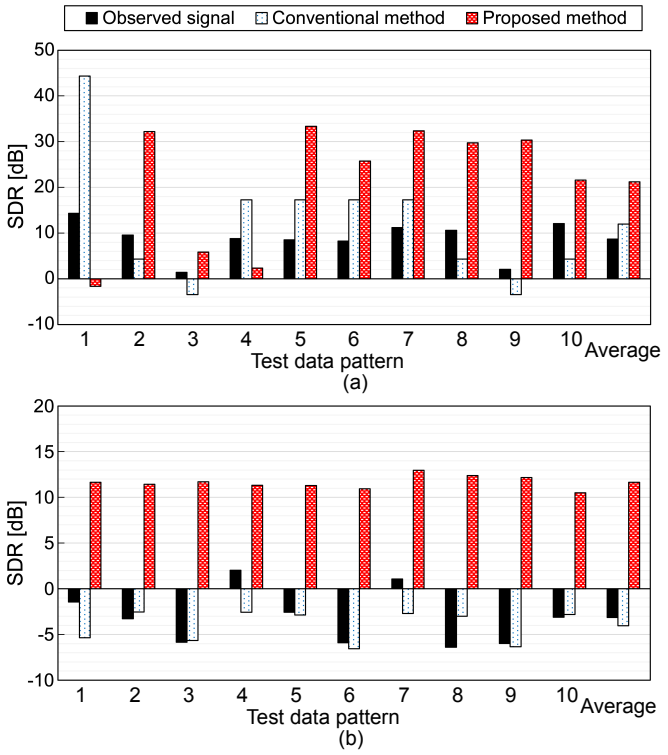


Fig. 8. Average SDR for out-domain test data: (a) music mixtures with speech models and (b) speech mixtures with music models.

and nonnegative matrix factorization," *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, vol. 24, no. 9, pp. 1626–1641, 2016.

[14] D. Kitamura, N. Ono, H. Sawada, H. Kameoka, and H. Saruwatari, "Determined blind source separation with independent low-rank matrix analysis," in *Audio Source Separation*, S. Makino, Ed., pp. 125–155. Springer, Cham, 2018.

[15] D. D. Lee and H. S. Seung, "Learning the parts of objects by non-negative matrix factorization," *Nature*, vol. 401, pp. 788–791, 1999.

[16] N. Makishima, S. Mogami, N. Takamune, D. Kitamura, H. Sumino, S. Takamichi, H. Saruwatari, and N. Ono, "Independent deeply learned matrix analysis for determined audio source separation," *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, vol. 27, no. 10, pp. 1601–1615, 2019.

[17] S. Oyabu, D. Kitamura, and K. Yatabe, "Linear multichannel blind source separation based on time-frequency mask obtained by harmonic/percussive sound separation," in *Proc. ICASSP*, pp. 201–205, 2021.

[18] K. Yatabe and D. Kitamura, "Determined BSS based on time-frequency masking and its application to harmonic vector analysis," *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, vol. 29, pp. 1609–1625, 2021.

[19] D. Kitamura, N. Ono, and H. Saruwatari, "Experimental analysis of optimal window length for independent low-rank matrix analysis," in *Proc. EUSIPCO*, pp. 1210–1214, 2017.

[20] Y. Luo and N. Mesgarani, "Conv-TasNet: Surpassing ideal time–frequency magnitude masking for speech separation," *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, vol. 27, no. 8, pp. 1256–1266, 2019.

[21] E. Tzinis, Z. Wang, and P. Smaragdis, "Sudo rm -rf: Efficient networks for universal audio source separation," *MLSP*, pp. 1–6, 2020,

[22] A. Défossez, "Hybrid spectrogram and waveform source separation," *Proc. ISMIR*, 2021.

[23] S. Yamaji and D. Kitamura, "DNN-based permutation solver for frequency-domain independent component analysis in two-source mixture case," *Proc. APSIPA ASC*, pp. 781–787, 2020.

[24] M. Kowalski, E. Vincent, and R. Gribonval, "Beyond the narrowband approximation: wideband convex methods for under-determined reverberant audio source separation," *IEEE Trans. Audio, Speech, Lang. Process.*, vol. 18, no. 7, pp. 1818–1829, 2010.

[25] A. Horn, "Doubly stochastic matrices and the diagonal of a rotation matrix," *Am. J. Math.*, vol. 76, no. 3, pp. 620–630, 1954.

[26] R. Sinkhorn and P. Knopp. "Concerning nonnegative matrices and doubly stochastic matrices," *Pacific J. Math.*, vol. 21, no. 2, pp. 343–348, 1967.

[27] K. Matsuoka and S. Nakashima, "Minimal distortion principle for blind source separation," in *Proc. ICA*, pp. 722–727, 2001.

[28] G. Birkhoff, "Three observations on linear algebra," *Univ. Nac. Tucuman, Rev. Ser. A.*, vol. 5, pp. 147–151, 1946.

[29] V. Nair and G. E. Hinton, "Rectified linear units improve restricted Boltzmann machines," in *Proc. ICML*, 2010.

[30] D. Yu, M. Kolbak, Z.-H. Tan, and J. Jensen, "Permutation invariant training of deep models for speaker-independent multi-talker speech separation," in *Proc. ICASSP*, pp. 241–245, 2017.

[31] S. Araki, F. Nesta, E. Vincent, Z. Koldovský, G. Nolte, A. Ziehe, and A. Benichoux, "The 2011 signal separation evaluation campaign (SiSEC2011):-audio source separation," in *Proc. LVA/ICA*, pp. 414–422, 2012.

[32] Y. Liang, S. M. Naqvi, and J. A. Chambers, "Overcoming block permutation problem in frequency domain blind source separation when using AuxIVA algorithm," *Electron. Lett.*, vol. 48, no. 8, pp. 460–462, 2012.

[33] E. Vincent, R. Gribonval, and C. Févotte, "Performance measurement in blind audio source separation," *IEEE Trans. Audio, Speech, Lang. Process.*, vol. 14, no. 4, pp. 1462–1469, 2006.