Low Latency Online Source Separation and Noise Reduction Based on Joint Optimization with Dereverberation

Tetsuya Ueda*, Tomohiro Nakatani†, Rintaro Ikeshita†, Keisuke Kinoshita†, Shoko Araki†, Shoji Makino*
*University of Tsukuba, 1-1-1 Tennodai, Tsukuba, Ibaraki 305-8577, Japan †NTT Corporation, Japan
Email: t.ueda@mmlab.cs.tsukuba.ac.jp, tomohiro.nakatani.nu@hco.ntt.co.jp, rintaro.ikeshita.fh@hco.ntt.co.jp, keisuke.kinoshita@ieee.org, shoko.araki.pu@hco.ntt.co.jp, maki@tara.tsukuba.ac.jp

***Abstract*—This paper proposes low latency online source separation in noisy environments. An approach based on weighted prediction error dereverberation was recently proposed to solve the degradation caused by using low latency online source separation. Although this approach can also reduce noise by increasing the number of microphones and separating the noise as additional sources, the calculation cost prohibitively increases. To solve this problem, this paper incorporates techniques used in independent vector extraction (IVE) into the above conventional approach. Because IVE can skip most of the calculations for estimating noise by assuming that it is a stationary Gaussian, our proposed method achieves effective and computationally efficient noise reduction using many microphones. Experiments in a noisy car environment show that our proposed online method simultaneously separates sources and reduces noise with low latency (< 12 ms) processing.**

***Index Terms*—Blind source separation, blind dereverberation, online, independent vector extraction, low latency**

I. Introduction

Speech enhancement is helpful for such applications as hearing aids and in-car communication systems, which transmit passengers' voices between the front and back seats for comfortable conversations in a vehicle [1]. For speech enhancement in such systems, we need to simultaneously address two major problems: 1) real-time processing with little delay, and 2) simultaneous reduction of interfering voices and background noise.

To solve the above problems, we use online blind source separation (BSS). Online BSS is a technique that separates individual source signals from current and past microphone array inputs without any prior information about the source signals. Independent vector analysis (IVA) [2] is a BSS approach that works in the frequency domain. An advantage of IVA is that it solves the frequency permutation problem without relying on any post-processing, assuming that each source has time-dependent and frequency-invariant magnitudes. Under this assumption, a fast online BSS algorithm with rapid convergence and low computational cost was successfully developed [3] based on an auxiliary-function-based IVA [4], [5].

To achieve low latency processing, cascading weighted prediction error (WPE) [6]–[8] based dereverberation with an online-IVA has been effective [9]. With reduced reverberation, we can substantially shorten the short time Fourier transform (STFT) frames without seriously degrading the BSS performance and greatly reduce the algorithmic delay [10]. Jointly optimizing IVA and WPE can also achieve substantially higher separation performance than simply cascading them [9], [11]–[13].

One drawback of the above method is that it does not consider background noise. We empirically expected to effectively reduce noise by increasing the number of microphones. However, the computational cost increases when the number of microphones is large, which limits the method's applicability to online processing. To solve this problem, we incorporate techniques used in a variant of IVA, namely, independent vector extraction (IVE) [14]–[21], into the low latency online BSS. By replacing IVA with IVE, we can optimally separate sources that are much fewer in number than the number of microphones and skip most of the calculation for estimating the noise statistics. As a result, the proposed method can achieve computationally efficient noise reduction using many microphones. Joint IVE and WPE optimization was recently presented for offline processing [22], [23]. In contrast, our work is the first to give an online algorithm for IVE [20] and to develop a method for optimizing it jointly with online-WPE. We then evaluate our method's performance by separation experiments on two-speaker eight-microphone mixtures and show that it effectively reduces background noise in real-time with low latency (< 12 ms) in a noisy car environment.

II. Problem formulation

Suppose that N sources and $M - N$ noises are captured by M microphones[1] and that the captured signals can be modeled at each time $t = 1, \ldots, T$ and frequency $f = 1, \ldots, F$ in the STFT domain:

$$\boldsymbol{x}(f,t) = \sum_{\tau=0}^{L_A-1} \boldsymbol{A}(f,\tau)\boldsymbol{s}(f,t-\tau), \tag{1}$$

where $\boldsymbol{x}(f,t) = [x_1(f,t), \ldots, x_M(f,t)]^{\mathsf{T}} \in \mathbb{C}^M$ is the vector containing the microphone signals, $\boldsymbol{s}(f,t) =$

[1]This assumption is introduced for algorithm derivation, and in practice the proposed method can perform noise reduction even in diffuse noise environments, as shown by our experiments.

TABLE I: Classification table of source separation.

	Offline		Online	
w/o WPE	IVA [5]	IVE [14], [15], [20]	Online-IVA [3]	Online-IVE **proposed**
w/ WPE	WPE×IVA [21]	WPE×IVE [22], [23]	Online-WPE×IVA [9]	Online-WPE×IVE **proposed**

$[s_1(f,t),\ldots,s_M(f,t)]^{\mathsf{T}} \in \mathbb{C}^M$ is that containing N source signals for $n \in \{1,\ldots,N\}$ and $M-N$ noise signals for $n \in \{N+1,\ldots,M\}$, where $(\cdot)^{\mathsf{T}}$ denote the transpose. $\boldsymbol{A}(f,\tau) \in \mathbb{C}^{M\times M}$ for $\tau = 0,\ldots,L_A-1$ are the convolutional transfer function matrices from the corresponding sources and noises to the microphones, where L_A is the order of convolution. Our aim here is to estimate a separation matrix that separates out individual source signals $s_1(f,t),\ldots,s_N(f,t)$ from $\boldsymbol{x}(f,t)$ in an online approach with a short STFT frame (= low algorithmic delay) for an over-determined case ($M \gg N$). Note that it is unnecessary to extract noise signals $s_{N+1}(f,t),\ldots,s_M(f,t)$.

III. Proposed online methods

We introduce our proposed methods by showing a classification table for source separation in Table I. Although online-IVA [3] and the online joint optimization of WPE and IVA (referred to as "online-WPE×IVA") [9] are already shown, the computational cost increases to enlarge the number of microphones for extracting target signals in a noisy environment. On the other hand in offline processing, IVE and joint optimization of WPE and IVE can extract only target signals with low calculation [15], [20], [22], [23]. However, their online algorithms have not been shown. In this section, we propose an online version of IVE ("online-IVE") and an online joint optimization of WPE and IVE ("online-WPE×IVE"). Since online-WPE×IVE is upwardly compatible with an online-IVE, in the remainder of this section, we mainly describe the model and optimization algorithm of online-WPE×IVE.

A. Models of beamformers and source signals

To derive online joint dereverberation, source separation, and noise reduction algorithms, we adopt almost the same models of beamformers and source signals as those previously used [22], [23], except for modifications required for coping with online processing. First, we assume that the relationship between $\boldsymbol{x}(f,t)$, and $\boldsymbol{s}(f,t)$ can be modeled using a convolutional beamformer (CBF):

$$\boldsymbol{s}(f,t) = \boldsymbol{W}^{\mathsf{H}}(f)\begin{pmatrix}\boldsymbol{x}(f,t)\\ \overline{\boldsymbol{x}}(f,t)\end{pmatrix} \in \mathbb{C}^M, \tag{2}$$

$$\overline{\boldsymbol{x}}(f,t) = [\boldsymbol{x}^{\mathsf{T}}(f,t-D),\ldots,\boldsymbol{x}^{\mathsf{T}}(f,t-D-L+1)]^{\mathsf{T}} \in \mathbb{C}^{ML}, \tag{3}$$

where $\boldsymbol{W}(f) \in \mathbb{C}^{M(L+1)\times M}$, is a coefficient matrix, $(\cdot)^{\mathsf{H}}$ denotes the Hermitian transpose, and $\overline{\boldsymbol{x}}(f,t)$ is a vector containing a past observation. L is the CBF length, and D is the prediction delay.

Similar to an offline IVE approach [23, Algorithm 2], we decompose convolutional filter $\boldsymbol{W}(f)$ into dereverberation matrix $\boldsymbol{G}(f) \in \mathbb{C}^{ML\times M}$ and separation matrix $\boldsymbol{Q}(f) \in \mathbb{C}^{M\times M}$:

$$\boldsymbol{y}(f,t) = \boldsymbol{x}(f,t) - \boldsymbol{G}^{\mathsf{H}}(f)\overline{\boldsymbol{x}}(f,t), \tag{4}$$

$$\boldsymbol{s}(f,t) = \boldsymbol{Q}^{\mathsf{H}}(f)\boldsymbol{y}(f,t), \tag{5}$$

where $\boldsymbol{W}^{\mathsf{H}}(f) = \boldsymbol{Q}^{\mathsf{H}}(f)[\boldsymbol{I}_M, -\boldsymbol{G}^{\mathsf{H}}(f)]$, $\boldsymbol{I}_M \in \mathbb{C}^{M\times M}$ is an identity matrix and $\boldsymbol{y}(f,t) \in \mathbb{C}^M$ is a dereverberated signal. With reduced reverberation in $\boldsymbol{y}(f,t)$, Eq. (5) can perform effective source separation even with short STFT frames.

Next, to derive an objective of the optimization, we introduce the same probabilistic source models used in IVE [20]:

$$p(\{s_n(f,t)\}_{n,f,t}) = \prod_{n,f,t} p(s_n(f,t)), \tag{6}$$

$$p(s_n(f,t)) = \mathcal{N}_{\mathbb{C}}(0, v_n(t)), \tag{7}$$

$$v_n(t) = 1, \text{ where } n \in \{N+1,\ldots,M\}. \tag{8}$$

Equation (6) specifies mutual independence between the sources, and $\mathcal{N}_{\mathbb{C}}(0,v_n(t))$ denotes a complex Gaussian distribution with a mean zero and variance $v_n(t)$. Noise signals are assumed to be stationary Gaussian signals.

Under the above assumptions, given past and current microphone signals $\mathcal{X}_t = \{x_m(f,t')\}_{f,t'\le t,m}$ with forgetting factor $0 < \beta < 1$, negative log-likelihood $\mathcal{I}$ becomes:

$$\mathcal{I}(\mathcal{X}_t) \stackrel{c}{=} -2\sum_f \log\left|\det \boldsymbol{Q}(f;t)\right| + \frac{1}{\sum_{t'\le t}\beta^{t-t'}}\sum_{f,t'\le t,n}\beta^{t-t'}\left(\log v_n(t') + \frac{|s_n(f,t')|^2}{v_n(t')}\right), \tag{9}$$

where $\stackrel{c}{=}$ denotes equality up to the constant terms, and $\boldsymbol{Q}(f;t)$ denotes the calculated $\boldsymbol{Q}(f)$ at time t.

B. Optimization by online approach

We use a recursive coordinate descent method to obtain a local optimal solution of Eq. (9). At each frame, after estimating $\boldsymbol{y}(f,t)$ and $\boldsymbol{s}(f,t)$ from $\boldsymbol{x}(f,t)$ based on Eqs. (4) and (5) using $\mathcal{G}_{t-1}$ and $\mathcal{Q}_{t-1}$ obtained in the previous frame, we recursively update $\mathcal{V}_t = \{v_n(t)\}_n$, $\mathcal{G}_t = \{\boldsymbol{G}(f;t)\}_f$, and $\mathcal{Q}_t = \{\boldsymbol{Q}(f;t)\}_f$, based on the following three minimization steps:

$$\mathcal{V}_t \leftarrow \operatorname*{argmin}_{\mathcal{V}_t} \mathcal{I}(\mathcal{X}_t;\mathcal{G}_{t-1},\mathcal{Q}_{t-1},\mathcal{V}_t), \tag{10}$$

$$\mathcal{G}_t \leftarrow \operatorname*{argmin}_{\mathcal{G}_t} \mathcal{I}(\mathcal{X}_t;\mathcal{G}_t,\mathcal{Q}_{t-1},\mathcal{V}_t), \tag{11}$$

$$\mathcal{Q}_t \leftarrow \operatorname*{argmin}_{\mathcal{Q}_t} \mathcal{I}(\mathcal{X}_t;\mathcal{G}_t,\mathcal{Q}_t,\mathcal{V}_t). \tag{12}$$

In the following, we describe the update equations for the above steps.

1) Update of $\mathcal{V}_t$: IVE solves the permutation alignment of the separated components in each frequency using a frequency-independent source variance model. With the model, the variances are updated at each time:

$$v_n(t) \leftarrow \frac{1}{F}\sum_{f=1}^{F}|s_n(f,t)|^2 \text{ for } n = 1,\ldots,N. \quad (13)$$

After calculating $v_n(t)$, we modify the amplitude of $s_n(f,t)$ by applying a back-projection technique [24]. In what follows, because we can independently update $\mathcal{G}_t$ and $\mathcal{Q}_t$ for each frequency bin, we drop frequency index f off to ease the notation.

2) Update of $\mathcal{G}_t$: To update $\mathcal{G}_t$, Eq. (9) can be rewritten:

$$\mathcal{I}(\mathcal{G}_t) = \sum_{n\in[1,M]} \|(\boldsymbol{G}(t) - \boldsymbol{R}_n^{-1}(t)\boldsymbol{P}_n(t))\boldsymbol{q}_n(t-1)\|^2_{\boldsymbol{R}_n(t)}, \quad (14)$$

where $\boldsymbol{R}_n(t) = \beta\boldsymbol{R}_n(t-1) + \overline{\boldsymbol{x}}(t)\overline{\boldsymbol{x}}^{\mathsf{H}}(t)/v_n(t)$ and $\boldsymbol{P}_n(t) = \beta\boldsymbol{P}_n(t-1) + \overline{\boldsymbol{x}}(t)\boldsymbol{x}^{\mathsf{H}}(t)/v_n(t)$ are spatio-temporal covariance matrices in a recursive form, $\boldsymbol{q}_n(t)$ is the nth column of $\boldsymbol{Q}(t)$, and $\|\boldsymbol{x}\|^2_{\boldsymbol{R}} = \boldsymbol{x}^{\mathsf{H}}\boldsymbol{R}\boldsymbol{x}$. Eq. (14) can be minimized when $\boldsymbol{G}(t)$ satisfies the following equation that was also previously shown [23, Algorithm 2]:

$$\boldsymbol{G}(t)\boldsymbol{q}_n(t-1) = \boldsymbol{R}_n^{-1}(t)\boldsymbol{P}_n(t)\boldsymbol{q}_n(t-1) \text{ for all } n. \quad (15)$$

Now, let $\boldsymbol{G}_n(t) = \boldsymbol{R}_n^{-1}(t)\boldsymbol{P}_n(t)$ for $n \in \{1,\ldots,N+1\}$. Then similar to online-WPE [7]–[9], $\boldsymbol{G}_n(t)$ can be recursively updated based on the matrix inversion lemma [25]:

$$\boldsymbol{K}_n(t) \leftarrow \frac{\boldsymbol{R}_n^{-1}(t-1)\overline{\boldsymbol{x}}(t)}{\beta v_n(t) + \overline{\boldsymbol{x}}^{\mathsf{H}}(t)\boldsymbol{R}_n^{-1}(t-1)\overline{\boldsymbol{x}}(t)}, \quad (16)$$

$$\boldsymbol{R}_n^{-1}(t) \leftarrow \{\boldsymbol{R}_n^{-1}(t-1) - \boldsymbol{K}_n(t)\overline{\boldsymbol{x}}^{\mathsf{H}}(t)\boldsymbol{R}_n^{-1}(t-1)\}/\beta, \quad (17)$$

$$\boldsymbol{G}_n(t) \leftarrow \boldsymbol{G}_n(t-1) + \boldsymbol{K}_n(t)\{\boldsymbol{x}(t) - \boldsymbol{G}_n^{\mathsf{H}}(t-1)\overline{\boldsymbol{x}}(t)\}^{\mathsf{H}}, \quad (18)$$

where $\boldsymbol{K}_n(t)$ is the Kalman gain vector. Because $\boldsymbol{G}_n(t)$ takes the same value for $n \in \{N+1,\ldots,M\}$, here we skip the calculation of $\boldsymbol{G}_n(t)$ for $n \in \{N+2,\ldots,M\}$. Then Eq. (15) can be written:

$$\boldsymbol{G}(t)\boldsymbol{Q}(t-1) = [\boldsymbol{G}_1(t)\boldsymbol{q}_1(t-1),\ldots,\boldsymbol{G}_N(t)\boldsymbol{q}_N(t-1), \boldsymbol{G}_{N+1}(t)\boldsymbol{Q}_{\mathrm{N}}(t-1)] \quad (19)$$

where $\boldsymbol{Q}_{\mathrm{N}}(t) = [\boldsymbol{q}_{N+1}(t),\ldots,\boldsymbol{q}_M(t)] \in \mathbb{C}^{M\times(M-N)}$. Therefore, Eq. (14) can be minimized in an online update:

$$\boldsymbol{G}(t) \leftarrow [\boldsymbol{G}_1(t)\boldsymbol{q}_1(t-1),\ldots,\boldsymbol{G}_N(t)\boldsymbol{q}_N(t-1), \boldsymbol{G}_{N+1}(t)\boldsymbol{Q}_{\mathrm{N}}(t-1)]\boldsymbol{Q}^{-1}(t-1). \quad (20)$$

3) Update of $\mathcal{Q}_t$: To update $\mathcal{Q}_t$, the log likelihood can be rewritten:

$$\mathcal{I}(\mathcal{Q}_t) \stackrel{c}{=} \sum_{n,f}\|\boldsymbol{q}_n(t)\|^2_{\boldsymbol{\Sigma}_n(t)} - 2\sum_f \log\left|\det \boldsymbol{Q}(t)\right|, \quad (21)$$

where $\boldsymbol{\Sigma}_n(t)$ is a covariance matrix used for optimization. Here we calculate $\boldsymbol{\Sigma}_n(t)$ for $n \in \{1,\ldots,N+1\}$ by following previous studies [3], [16]:

$$\boldsymbol{\Sigma}_n(t) \leftarrow \alpha\boldsymbol{\Sigma}_n(t-L_b) + \frac{(1-\alpha)}{L_b}\sum_{\tau=t-L_b+1}^{t}\frac{\boldsymbol{y}(\tau)\boldsymbol{y}^{\mathsf{H}}(\tau)}{v_n(\tau)}, \quad (22)$$

where we slightly modify the configuration of the recursive update from Eq. (9), by setting a different forgetting factor $0 < \alpha < 1$ and introducing a block-based covariance update with block length L_b. Because likelihood function (21) is identical to that for a conventional IVE technique [15], [20], we apply the same computationally efficient update equations as IVE does. After initializing $\boldsymbol{Q}(t) = \boldsymbol{Q}(t-1)$ at each frame, we update $\boldsymbol{q}_n(t)$ using iterative projection [5] in each $n \in \{1,\ldots,N\}$:

$$\boldsymbol{q}_n(t) \leftarrow (\boldsymbol{Q}^{\mathsf{H}}(t)\boldsymbol{\Sigma}_n(t))^{-1}\boldsymbol{e}_n, \quad (23)$$

$$\boldsymbol{q}_n(t) \leftarrow \frac{\boldsymbol{q}_n(t)}{\sqrt{\boldsymbol{q}_n^{\mathsf{H}}(t)\boldsymbol{\Sigma}_n(t)\boldsymbol{q}_n(t)}}, \quad (24)$$

where $\boldsymbol{e}_n$ denotes the nth column of $\boldsymbol{I}_M$. Then we update the noise part of filter $\boldsymbol{Q}(t)$ in the same way as previous work [15], [20]:

$$\boldsymbol{Q}_{\mathrm{N}}(t) \leftarrow \begin{pmatrix} -(\boldsymbol{Q}_{\mathrm{S}}^{\mathsf{H}}(t)\Sigma_{N+1}(t)\boldsymbol{E}_N)^{-1}(\boldsymbol{Q}_{\mathrm{S}}^{\mathsf{H}}(t)\Sigma_{N+1}(t)\boldsymbol{E}_{M-N}) \\ \boldsymbol{I}_{M-N} \end{pmatrix}, \quad (25)$$

where $\boldsymbol{Q}_{\mathrm{S}}(t) = [\boldsymbol{q}_1(t),\ldots,\boldsymbol{q}_N(t)] \in \mathbb{C}^{M\times N}$, $\boldsymbol{E}_N$ and $\boldsymbol{E}_{M-N}$ are the first N and the remaining $M-N$ columns of $\boldsymbol{I}_M$. Since $\boldsymbol{Q}_{\mathrm{N}}(t)$ can be obtained by a single step update that doesn't depend on the number of microphones, we can largely reduce the computational cost especially when we have many microphones.

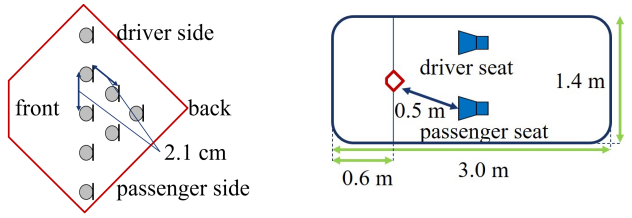
4) Summary: The algorithm of online-WPE×IVE in each t, f, and n is composed of the following four steps:

1) Calculate $\boldsymbol{y}(f,t)$, and $\boldsymbol{s}(f,t)$, using Eqs. (4) and (5).
2) Update $v_n(t)$ using Eq. (13).
3) Update $\boldsymbol{G}(f;t)$ using Eqs. (16)–(18), Eq. (20).
4) Update $\boldsymbol{q}_n(f;t)$ using Eqs. (22)–(25).

We can easily implement online-IVE by treating $\boldsymbol{G}(f)$ in Eq. (4) as a zero matrix and skipping the update of $\boldsymbol{G}(f)$ in Eqs. (16)–(18). This algorithm is novel for our paper.

IV. Experimental evaluation

We evaluated the effectiveness of our proposed method, by conducting a source separation experiment in a car with $N = 2$ and $M = 8$. We used set B of the ATR digital speech database [26], which is composed of speech data from ten speakers (six men and four women). By randomly selecting two different speakers from the database and iterating each utterance so that each signal is 20 seconds, we obtained 100 pairs of source signals. Mixture signals $u_m(f,t)$ for $m \in \{1,\ldots,M\}$ are generated by mixing source signals $[s_1(f,t),\ldots,s_N(f,t)]$ after being convoluted with impulse responses. We measured the impulse responses using a time-stretched pulse in a car.



(a) Microphones (b) View inside car

Fig. 1: Experimental sound source and microphone layout

The recording environment is shown in Fig. 1. We set one speaker on the driver's seat, another on the front passenger's seat, and microphones on a map lamp in the car. Reverberation time[2] T_{60} in the car was 48 ms. Noise signals $z_m(f,t)$ for $m \in \{1,\ldots,M\}$ were recorded in a running car. Then we generated microphone signals $\boldsymbol{x}(f,t)$ by adding mixture signals $\boldsymbol{u}(f,t) = [u_1(f,t),\ldots,u_M(f,t)]^{\mathsf{T}}$ and noise signals $\boldsymbol{z}(f,t) = [z_1(f,t),\ldots,z_M(f,t)]^{\mathsf{T}}$ so that the input signal-to-noise ratio (input-SNR) $= 10\log_{10}\frac{\sum_t |z_1(f,t)|^2}{\sum_t |u_1(f,t)|^2}$ dB becomes a specified value. The sampling rate was 16 kHz.

In the experiment, we compared four online-methods: online-WPE×IVA [9], online-IVE, online-WPE+IVE, and online-WPE×IVE. Hereafter, we abbreviate "online-" to reduce redundancy. WPE+IVE can be implemented by cascading IVE after WPE [8]. WPE×IVA is equal to WPE×IVE with N changed from 2 to 8 (= M). Because WPE×IVA outputs 8 signals including noises, we selected 2 target signals using oracle information, i.e., based on correlation with the reference signals. We set the frame length and the shift at 8 ms and 4 ms based on the upper limit of the respective mouth-to-ear delays (= 12 ms) [27]. We took the average of the source-to-distortion ratio (SDR), the source-to-interference ratio (SIR), and the sources-to-noise ratio (SNR) as evaluation criteria [28]. In this paper, we used source signals $s_n(f,t)$ as reference for SDR and SIR. We used a square root Hanning window for the STFT window by computational efficiency. We set the other parameters $\{\alpha,\ \beta,\ L_b,\ D,\ L\}$ at $\{0.96,\ 0.9999,\ 2,\ 1,\ 4\}$, respectively. We initialized $\boldsymbol{Q}(f;0)$ and $\boldsymbol{R}_n^{-1}(f;0)$ as identity matrices, $\boldsymbol{G}(f;0)$, and $\boldsymbol{G}_n(f;0)$ as a zero matrix, and $\Sigma_n(f;0)$ as $10^{-5} \times \boldsymbol{I}_M$.

First, we evaluated the computational efficiency of our proposed method. Table II shows the mean computation times for 20 s of 8-ch input using python 3.7.7 on a computer with an Intel Xeon Gold 2.4 GHz 1-core CPU. The total delay of WPE×IVE was 11.8 ms ($<$ 12 ms), including the algorithmic delay (= 8 ms). Comparing WPE×IVA and WPE×IVE, the latter reduced the calculation time from 5.5×10^4 s to 1.9×10^4 s, which results in sufficient real-time operation of WPE×IVE.

Next we evaluated the separation performance of the proposed method. Table III compares the SDR improvements, SIR improvements, and SNR with each input-SNR and the online method. The improvement of SIR shows how much interference was suppressed and SNR shows how much noise was included. The SDR improvement is an overall evaluation value that includes SIR and SNR. We treat WPE×IVA as a reference because its real-time factor greatly exceeded 1.0 and it used oracle information for distinguishing the targets from noise. When comparing WPE×IVE and WPE×IVA (w/ oracle), the former achieved not necessarily higher but almost comparable separation performance with much faster calculations without post-processing. This is the advantage of our proposed WPE×IVE. In addition, when comparing all the IVE-based methods in all input-SNRs, we found no statistical difference in the SIR improvement. However, there was a significant difference in SDR improvement and SNR between WPE×IVE and the other methods. In particular, the proposed WPE×IVE reduced the SNR from -10 (= input-SNR) to -2.94 dB. This result suggests that using variance $v_n = 1$ for noises was effective for distinguishing the noise in the microphone signal.

Finally, we show the convergence curve of the separation performance. Fig. 2 compares the SDR improvements over time for each online method. To observe the variation in the SDRs over time, we calculated the SDRs every two seconds without signals overlap. Our proposed method provided the highest SDR improvement over almost all 20 seconds. Although the difference of the SDR improvement between WPE+IVE and WPE×IVE became smaller over time, there was significant difference in the first two seconds in all the input-SNRs.

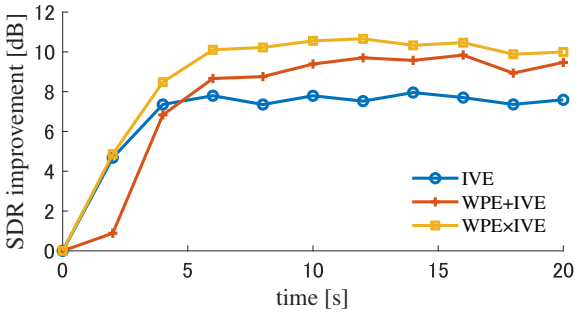
[2]This is much longer than an STFT frame in our experiments.

TABLE II: computational efficiency and total delay in separating 20-second signals (≒ 5000 frame)

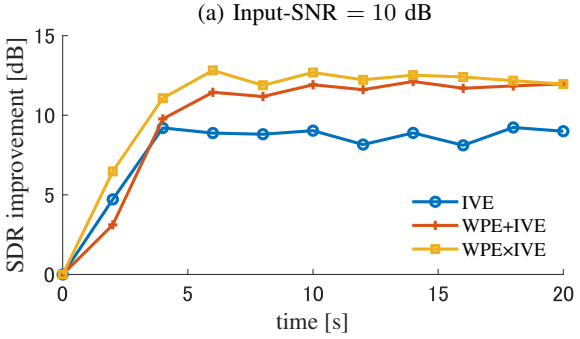
Time [ms] \ Online-methods	WPE×IVA	WPE×IVE
Calculation time	5.5×10^4	1.9×10^4
Calculation time for a frame (= ①)	11.0	3.8
Algorithmic delay (= ②)	8.0	
Total delay (= ① + ②)	19.0	11.8
Limit of delays [27]	12.0	

TABLE III: Average SDR, SIR improvement, and SNR for each online method: Scores in parentheses are 1.96×standard error. Bold font shows top scores. † and †† mean that top score method has a statistical difference from other methods at 10% and 5% levels.

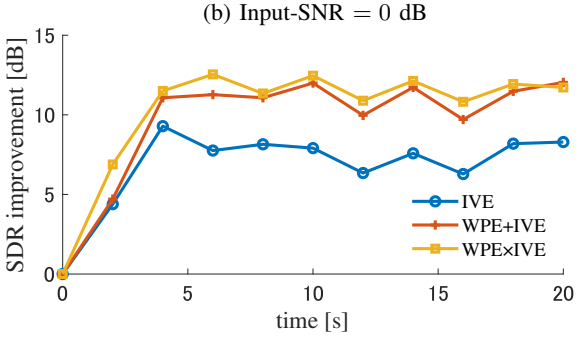
online-methods	SDR imp. [dB]	SIR imp. [dB]	SNR [dB]
Input-SNR = 10 [dB]			
WPE×IVA (w/ oracle)	8.23 (0.43)	23.60 (0.51)	7.16 (0.37)
IVE	5.74 (0.35)	21.41 (0.72)	4.89 (0.29)
WPE+IVE	6.12 (0.29)	21.65 (0.55)	5.07 (0.25)
WPE×IVE	**7.61 (0.26)**††	**22.26 (0.43)**	**6.42 (0.25)**††
Input-SNR = 0 [dB]			
WPE×IVA (w/ oracle)	10.23 (0.48)	21.25 (0.47)	4.28 (0.42)
IVE	7.58 (0.31)	**21.85 (0.55)**	1.51 (0.28)
WPE+IVE	9.43 (0.29)	21.66 (0.43)	3.30 (0.28)
WPE×IVE	**10.77 (0.28)**††	21.19 (0.31)	**4.63 (0.27)**††
Input-SNR = -10 [dB]			
WPE×IVA (w/ oracle)	10.32 (0.59)	17.18 (0.51)	-4.00 (0.52)
IVE	7.42 (0.32)	18.48 (0.43)	-7.16 (0.3)
WPE+IVE	10.51 (0.35)	**18.99 (0.37)**	-4.15 (0.34)
WPE×IVE	**11.74 (0.32)**†	18.52 (0.28)	**-2.94 (0.32)**†



(a) Input-SNR = 10 dB



(b) Input-SNR = 0 dB



(c) Input-SNR = −10 dB

Fig. 2: Separation performance for each online method.

V. Conclusion

We proposed low latency online blind source separation in noisy environment. This method jointly optimized dereverberation, separation, and noise reduction with little algorithmic delay (= 8 ms) and low computational cost per frame (= 3.8 ms). The experimental results confirmed that our proposed online method reduced SNR from -10 to -2.94 dB with high source separation performance in-car environments.

References

[1] R. Landgraf, J. Köhler-Kaeß, C. Lüke, O. Niebuhr, and G. Schmidt, "Can you hear me now? Reducing the lombard effect in a driving car using an in-car communication system," in *Proc. Speech Prosody*, 2016, pp. 479–483.

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

[3] T. Taniguchi, N. Ono, A. Kawamura, and S. Sagayama, "An auxiliary-function approach to online independent vector analysis for real-time blind source separation," in *Proc. HSCMA*, 2014, pp. 107–111.

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

[5] N. Ono and S. Miyabe, "Auxiliary-function-based independent component analysis for super-Gaussian sources," in *Proc. LVA/ICA*, 2010, pp. 165–172.

[6] T. Nakatani, T. Yoshioka, K. Kinoshita, M. Miyoshi, and B.-H. Juang, "Blind speech dereverberation with multi-channel linear prediction based on short time Fourier transform representation," in *Proc. ICASSP*, 2008, pp. 85–88.

[7] T. Yoshioka, H. Tachibana, T. Nakatani, and M. Miyoshi, "Adaptive dereverberation of speech signals with speaker-position change detection," in *Proc. ICASSP*, 2009, pp. 3733–3736.

[8] J. Caroselli, I. Shafran, A. Narayanan, and R. Rose, "Adaptive multichannel dereverberation for automatic speech recognition," in *Proc. Interspeech*, 2017, pp. 3877–3881.

[9] T. Ueda, T. Nakatani, R. Ikeshita, K. Kinoshita, S. Araki, and S. Makino, "Low latency online blind source separation based on joint optimization with blind dereverberation," in *Proc. ICASSP*, 2021, pp. 506–510.

[10] D. Mauler and R. Martin, "A low delay, variable resolution, perfect reconstruction spectral analysis-synthesis system for speech enhancement," in *Proc. EUSIPCO*, 2007, pp. 222–226.

[11] T. Yoshioka, T. Nakatani, M. Miyoshi, and H.G. Okuno, "Blind separation and dereverberation of speech mixtures by joint optimization," *IEEE Trans. ASLP*, vol. 19, no. 1, pp. 69–84, 2010.

[12] H. Kagami, H. Kameoka, and M. Yukawa, "Joint separation and dereverberation of reverberant mixtures with determined multichannel non-negative matrix factorization," in *Proc. ICASSP*, 2018, pp. 31–35.

[13] T. Nakatani, C. Boeddeker, K. Kinoshita, R. Ikeshita, M. Delcroix, and R. Haeb-Umbach, "Jointly optimal denoising, dereverberation, and source separation," *IEEE/ACM Trans. ASLP*, vol. 28, pp. 2267–2282, 2020.

[14] Z. Koldovsky and P. Tichavsky, "Gradient algorithms for complex non-Gaussian independent component/vector extraction, question of convergence," *IEEE Trans. Signal Processing*, vol. 67, no. 4, pp. 1050–1064, 2018.

[15] R. Scheibler and N. Ono, "Independent vector analysis with more microphones than sources," in *Proc. WASPAA*, 2019, pp. 185–189.

[16] J. Jansky, J. Malek, J. Cmejla, T. Kounovsky, Z. Koldovsky, and J. Zdansky, "Adaptive blind audio source extraction supervised by dominant speaker identification using x-vectors," in *Proc. ICASSP*, 2020, pp. 676–680.

[17] R. Scheibler and N. Ono, "Fast independent vector extraction by iterative SINR maximization," in *Proc. ICASSP*, 2020, pp. 601–605.

[18] R. Scheibler and N. Ono, "MM algorithms for joint independent subspace analysis with application to blind single and multi-source extraction," *arXiv preprint arXiv:2004.03926*, 2020.

[19] R. Ikeshita, T. Nakatani, and S. Araki, "Block coordinate descent algorithms for auxiliary-function-based independent vector extraction," *IEEE Trans. Signal Processing*, 2021.

[20] R. Ikeshita, T. Nakatani, and S. Araki, "Overdetermined independent vector analysis," in *Proc. ICASSP*, 2020, pp. 591–595.

[21] T. Nakatani, R. Ikeshita, K. Kinoshita, H. Sawada, and S. Araki, "Computationally efficient and versatile framework for joint optimization of blind speech separation and dereverberation," in *Proc. Interspeech*, 2020.

[22] T. Nakatani, R. Ikeshita, K. Kinoshita, H. Sawada, and S. Araki, "Blind and neural network-guided convolutional beamformer for joint denoising, dereverberation, and source separation," in *Proc. ICASSP*, 2021, pp. 6129–6133.

[23] R. Ikeshita and T. Nakatani, "Independent vector extraction for fast joint blind source separation and dereverberation," *IEEE Signal Processing Letters*, 2021.

[24] 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.

[25] S. Haykin, *Adaptive filter theory*, Pearson Education India, 2008.

[26] A. Kurematsu, K. Takeda, Y. Sagisaka, S. Katagiri, H. Kuwabara, and K. Shikano, "ATR japanese speech database as a tool of speech recognition and synthesis," *Speech communication*, vol. 9, no. 4, pp. 357–363, 1990.

[27] A. Theiss, G. Schmidt, J. Withopf, and C. Lueke, "Instrumental evaluation of in-car communication systems," in *Speech Communication; 11. ITG Symposium*. VDE, 2014, pp. 1–4.

[28] E. Vincent, R. Gribonval, and C. Févotte, "Performance measurement in blind audio source separation," *IEEE/ACM Trans. ASLP*, vol. 14, no. 4, pp. 1462–1469, 2006.