Journal of Signal Processing, Vol.27, No.4, pp.81-85, July 2023

SELECTED PAPER AT NCSP'23

# Real-Time Moving Blind Source Extraction Based on Constant Separating Vector and Auxiliary Function Technique

Sihan Yuan, Tetsuya Ueda and Shoji Makino

Graduate School of Information, Production and Systems, Waseda University
2-7 Hibikino, Wakamatsu-ku, Kitakyushu, Fukuoka 808-0135, Japan
E-mail: {yuansihan@fuji., t.ueda@akane., s.makino@}waseda.jp

**Abstract**

We propose a novel online algorithm for moving blind source extraction (BSE). The original algorithm is based on the recently proposed constant separating vector mixing model with batch auxiliary-function-based independent vector extraction (CSV-AuxIVE). The CSV-AuxIVE is not suitable for some devices that require real-time processing. In this case, we propose the online-CSV-AuxIVE, which only needs to know part of the mixed signal to process the observed signal sequentially. Then, we verified the effectiveness of the proposed method on a specific task of extracting moving two-speaker signals. In the experiment, the online-CSV-AuxIVE is compared with the online-AuxIVA. The result shows that under the same conditions of the source of interest (SOI), the average source-to-distortion ratio (SDR) of online-CSV-AuxIVE is approximately 1.5 dB higher than that of online-AuxIVA regardless of the changes in the range and speed of the interference (IR).

## 1. Introduction

As technology advances, an increasing number of speech devices are used in various practical scenarios, such as hearing aids, speech recognition modules in smart homes, and speech enhancement modules in online conferencing. All of these applications require extracting a specific source signal from the mixed signals recorded by the sensors, provided that the information about source signals is unknown. This technique is blind source extraction (BSE), or more generally, blind source separation (BSS). It has become one of the research hotspots in signal processing.

Many researchers have proposed numerous effective solutions. Among them, the most classical method is independent component analysis (ICA) [1, 2], in which it is assumed that the source signal is statistically independent and source separation is accomplished by maximizing non-Gaussianity. Alternatively, among the frequency domain-based methods, the most commonly used is independent vector analysis (IVA) [3, 4], as an extension of ICA, in which a joint statistical source model is used to treat all frequencies simultaneously. For faster convergence, IVA based on auxiliary functions (AuxIVA) [5,6] has been proposed, in which the effects of tuning parameters on the convergence speed are eliminated by introducing auxiliary functions that guarantee the monotonic decrease in the objective function. However, the above methods perform better with static systems (meaning all sources are stationary). In a realistic situation, the speakers will move back and forth. Thus, in such dynamic cases, these methods can be employed adaptively within small intervals during which the mixture is approximately stationary. In [7], researchers extended the batch-processing AuxIVA to an online process for the autoregressive approximation of auxiliary variables. Although this online method can adapt the speaker's movement, the accuracy is only sometimes so good because the target source's movement range and speed change. Significantly the performance will degrade when the source moves very quickly.

In recent years, the constant separating vector (CSV) mixing model [8] has been proposed for moving BSE. It involves both time-varying mixing and time-invariant separation parameters, effectively maintaining the extraction performance regardless of the speaker's movement. Moreover, this model is combined with auxiliary function-based independent vector extraction (AuxIVE) to accelerate the convergence. However, CSV-AuxIVE [9] is a batch processing algorithm, which is unsuitable for some devices requiring real-time processing, such as hearing aids and intelligent speech assistants. In this paper, we propose an online version of the CSV-AuxIVE model. Unlike the batch processing algorithm, the proposed algorithm only needs to know part of the mixed speech before the beginning of extraction to process the input observed signals sequentially. Finally, we evaluated the performance of the proposed algorithm in the experiment on dynamic two-speaker movements by comparing it with online-AuxIVA [7].

## 2. Conventional Method: CSV-AuxIVE

### 2.1 Blind source extraction

Let us first explain the BSS scheme assuming that there are $N$ sources recorded by $M$ microphones. After the short-time Fourier transform (STFT), let $s^n_{k,l}$ and $x^m_{k,l}$ denote the STFT coefficients of the $n$-th source and $m$-th microphone, where

$k = 1, \ldots, K$ and $l = 1, \ldots, L$ represent the frequency bin and time frame, respectively. In the frequency-domain instantaneous mixture model, the relationship between the observed signal $\boldsymbol{x}_{k,l} = [x^1_{k,l}, \ldots, x^M_{k,l}]^\mathsf{T} \in \mathbb{C}^M$ and the source signal $\boldsymbol{s}_{k,l} = [s^1_{k,l}, \ldots, s^N_{k,l}]^\mathsf{T} \in \mathbb{C}^N$ can be written as

$$\boldsymbol{x}_{k,l} = \boldsymbol{A}_{k,l}\boldsymbol{s}_{k,l} \tag{1}$$

where $\boldsymbol{A}_{k,l}$ stands for the $M \times N$ *mixing matrix* and $(\cdot)^\mathsf{T}$ denotes the transpose. If $M$ is equal to $N$, the inverse matrix of $\boldsymbol{A}_{k,l}$ exists and is called the *separation matrix* ($\boldsymbol{W}_{k,l}$), so Eq.(1) can be written as

$$\boldsymbol{s}_{k,l} = \boldsymbol{W}_{k,l}\boldsymbol{x}_{k,l} \tag{2}$$

For BSE, the task changes from separating each source to extracting a source of interest (SOI), so Eq.(1) can be rewritten as

$$\boldsymbol{x}_{k,l} = \boldsymbol{a}_{k,l}s_{k,l} + \boldsymbol{y}_{k,l} \tag{3}$$

where $s_{k,l}$ represents the SOI and $\boldsymbol{y}_{k,l}$ represents the remaining part of the mixture, which are independent of each other. In general, we assume that the SOI corresponds to the first source in $\boldsymbol{s}_{k,l} = [s^1_{k,l}, \ldots, s^N_{k,l}]^\mathsf{T} \in \mathbb{C}^N$, implying that the aboved source signal can be divided into $\boldsymbol{s}_{k,l} = [s_{k,l}, \boldsymbol{z}^\mathsf{T}_{k,l}]^\mathsf{T}$. Consequently, the *mixing vector* $\boldsymbol{a}_{k,l}$ is the first column vector in $\boldsymbol{A}_{k,l}$. Correspondingly, the *separation vector* $\boldsymbol{w}_{k,l}$ associated with the SOI is the first row in $\boldsymbol{W}_{k,l}$, which ensures $s_{k,l} = \boldsymbol{w}^\mathsf{H}_{k,l}\boldsymbol{x}_{k,l}$. $(\cdot)^\mathsf{H}$ denotes the Hermitian transpose. In this case, we limit the dimension of $\boldsymbol{y}_{k,l}$ to a subspace of dimension $M-1$, so $\boldsymbol{y}_{k,l}$ cannot be separated individually.

### 2.2 CSV mixing modeling

In a real situation, the mixing system is time-varying because of moving sources. Let us divide the observed signal into $T$ short intervals (blocks) based on time frames $L$. The length of each block is $L_b$, which satisfies $L_b = L/T$. Each block can be denoted by $t \in \{1, \ldots, T\}$. In other words, for the observed signal of each block, we consider the source is approximately to be stationary and compute the corresponding mixing vector $\boldsymbol{a}_{k,t}$ and separation vector $\boldsymbol{w}_{k,t}$.

In the CSV mixing model, we separate the mixing vector $\boldsymbol{a}_{k,l}$ from the mixing matrix $\boldsymbol{A}_{k,l}$ and can derive an extremely effective parameterization involving only the mixing vector associated with the SOI. The model considers only the mixing vector $\boldsymbol{a}_{k,l}$ to be block-dependent ($\boldsymbol{a}_{k,t}$) while the separation filter $\boldsymbol{w}_{k,l}$ is constant over the blocks ($\boldsymbol{w}_k$). Specifically, for the $t$-th block, the mixing matrix $\boldsymbol{A}_{k,t}$ in Eq.(1) has the following structure [10] (we omit frame index $l = 1, \ldots, L$ from here):

$$\boldsymbol{A}_{k,t} = [\boldsymbol{a}_{k,t}, \boldsymbol{D}_{k,t}] = \begin{bmatrix} \gamma_{k,t} & \boldsymbol{h}^\mathsf{H}_k \\ \boldsymbol{g}_{k,t} & \frac{1}{\gamma_{k,t}}(\boldsymbol{g}_{k,t}\boldsymbol{h}^\mathsf{H}_k - \boldsymbol{I}_{M-1}) \end{bmatrix} \tag{4}$$

Similarly, the structure of separation matrix $\boldsymbol{W}_{k,t}$ is

$$\boldsymbol{W}_{k,t} = \begin{bmatrix} \boldsymbol{w}^\mathsf{H}_k \\ \boldsymbol{B}_{k,t} \end{bmatrix} = \begin{bmatrix} \beta^*_k & \boldsymbol{h}^\mathsf{H}_k \\ \boldsymbol{g}_{k,t} & -\gamma_{k,t}\boldsymbol{I}_{M-1} \end{bmatrix} \tag{5}$$

where $\gamma_{k,t}$ is the scalar, $\boldsymbol{I}_M$ denotes the $M \times M$ identity matrix, the mixing vector $\boldsymbol{a}_{k,t}$ can be divided into $\boldsymbol{a}_{k,t} = [\gamma_{k,t}, \boldsymbol{g}^\mathsf{T}_{k,t}]^\mathsf{T}$, and the separation vector $\boldsymbol{w}_k$ can be divided into $\boldsymbol{w}_k = [\beta_k, \boldsymbol{h}^\mathsf{T}_k]^\mathsf{T}$. The CSV mixing model assumes that the mixing vector $\boldsymbol{a}_{k,t}$ and the distribution of sources can vary block by block, but the separation vector $\boldsymbol{w}_k$ is constant between blocks. Combining the above equations and the relationship between the mixing matrix $\boldsymbol{A}_{k,t}$ and the separation matrix $\boldsymbol{W}_{k,t}$, it is easy to deduce that the relationship between the mixing vector $\boldsymbol{a}_{k,t}$ and the separation vector $\boldsymbol{w}_k$ satisfies $\boldsymbol{w}^\mathsf{H}_k\boldsymbol{a}_{k,t} = 1$ (called distortionless constraint), which means that

$$\beta^*_k\gamma_{k,t} + \boldsymbol{h}^\mathsf{H}_k\boldsymbol{g}_{k,t} = 1 \tag{6}$$

In addition, the matrix $\boldsymbol{B}_{k,t}$ is called a *block matrix* [9], which satisfies $\boldsymbol{B}_k\boldsymbol{a}_k = \boldsymbol{0}$. The remaining signal source can be represented by $\boldsymbol{z}_k = \boldsymbol{B}_k\boldsymbol{x}_k = \boldsymbol{B}_k\boldsymbol{y}_k$, where $\boldsymbol{y}_k = \boldsymbol{D}_k\boldsymbol{z}_k$. Therefore, for the $t$-th block, Eq.(1) can be rewritten as

$$\boldsymbol{x}_{k,t} = \begin{bmatrix} \gamma_{k,t} & \boldsymbol{h}^\mathsf{H}_k \\ \boldsymbol{g}_{k,t} & \frac{1}{\gamma_{k,t}}(\boldsymbol{g}_{k,t}\boldsymbol{h}^\mathsf{H}_k - \boldsymbol{I}_{M-1}) \end{bmatrix} \begin{bmatrix} \boldsymbol{s}_{k,t} \\ \boldsymbol{z}_{k,t} \end{bmatrix} \tag{7}$$

In the CSV-AuxIVE algorithm, the auxiliary function approach [5, 6] was used to update the separation filter $\boldsymbol{w}_k$ efficiently. Here, the filter $\boldsymbol{w}_k$ is updated with several iterations as follows

$$\boldsymbol{w}_k = \left(\sum_{t=1}^{T} \frac{\boldsymbol{V}_{k,t}}{\hat{\sigma}^2_{k,t}}\right)^{-1} \sum_{t=1}^{T} \frac{\boldsymbol{w}^\mathsf{H}_k\boldsymbol{V}_{k,t}\boldsymbol{w}_k}{\hat{\sigma}^2_{k,t}}\boldsymbol{a}_{k,t} = \boldsymbol{P}^{-1}_k\boldsymbol{Q}_k \tag{8}$$

where $\hat{\sigma}^2_{k,t}$ denotes the variance of the source for the $t$-th block and $\boldsymbol{V}_{k,t}$ is the weighted spatial covariance matrix.

## 3. Proposed Method: Online-CSV-AuxIVE

In [9], although the authors divide the observed signal into several blocks in the frequency domain, the method does not process each block separately. This means that we must know all of the observed signals before extraction, which is the case of the original CSV-AuxIVE algorithm. However, for some devices that require the real-time processing of signals, such as hearing aids and multiperson online conferencing devices, the CSV-AuxIVE algorithm may lead to high latency and is not applicable. Therefore, to further improve the applicability of this algorithm, we propose an online version of the CSV-AuxIVE algorithm, with the specific aim of processing each piece of data sequentially and updating it to obtain a different mixing vector $\boldsymbol{a}_{k,t}$ for each block. In this paper, we call the

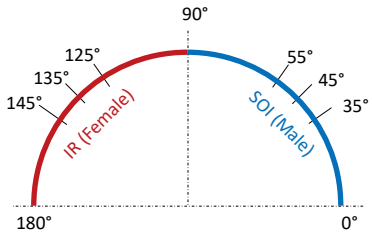



Figure 1: Layout of experimental environment

Table 1: Conditions of sources (range and speed)

| Index | Speed | Range (start angle → end angle) | | |
|---|---|---|---|---|
| | | | SOI (male) | IR (female) |
| 1 | Static | N/A | 45° | 135° |
| 2 | Slow | Small | 35° → 55° | 125° → 145° |
| 3 | Fast | Small | 35° → 55° | 125° → 145° |
| 4 | Slow | Large | 0° → 90° | 90° → 180° |
| 5 | Fast | Large | 0° → 90° | 90° → 180° |
| 6 | Very Fast | Large | 0° → 90° | 90° → 180° |

batch processing algorithm the offline CSV-AuxIVE model and the proposed algorithm the online model.

Moreover, we introduce the recursive update in each block to calculate $\boldsymbol{P}_{k,t}$ and $\boldsymbol{Q}_{k,t}$ as

$$\boldsymbol{P}_{k,t} = \alpha \boldsymbol{P}_{k,t-1} + (1-\alpha)\frac{\boldsymbol{V}_{k,t}}{\hat{\sigma}^2_{k,t}} \tag{9}$$

$$\boldsymbol{Q}_{k,t} = \alpha \boldsymbol{Q}_{k,t-1} + (1-\alpha)\frac{\boldsymbol{w}^{\mathsf{H}}_{k,t}\boldsymbol{V}_{k,t}\boldsymbol{w}_{k,t}}{\sigma^2_{k,t}}\boldsymbol{a}_{k,t} \tag{10}$$

where $\alpha$ is the forgetting factor. $\boldsymbol{P}_{k.t}$ and $\boldsymbol{Q}_{k,t}$ are updated after several iterations. The pseudo code is summarized in Algorithm 1. For the proposed algorithm, there are three parameters that have a large impact on the results: the forgetting factor $\alpha$, the block length $L_b$, and the initial value setting of the matrix $\boldsymbol{Q}_k$.

**Algorithm 1** Pseudocode of online-CSV-AuxIVE

**Input:** $\boldsymbol{x}_{k,t}$, $L_b$, $\alpha$, Maxiter
**Output:** $\boldsymbol{y}_{k,t}$

(Initialization)
$\boldsymbol{w}_{k,0} = [1, 0, ..., 0]^{\mathsf{T}} \in \mathbb{C}^{M\times N}$
$\boldsymbol{P}_{k,0} = \boldsymbol{R}_{k,0} = 10^{-12} \times$ Identity Matrix $\in \mathbb{C}^{M\times M}$
$\boldsymbol{Q}_{k,0} =$ Zero Matrix $\in \mathbb{C}^{M\times N}$
**for** $t = 1 \to T$ **do**
  (Enhancement)
  $\boldsymbol{y}_{k,t} = \boldsymbol{w}^{\mathsf{H}}_{k,t-1}\boldsymbol{x}_{k,t}$
  $[\gamma_{k,t-1}, \boldsymbol{g}^{\mathsf{T}}_{k,t-1}]^{\mathsf{T}} = \boldsymbol{a}_{k,t-1}$
  $y_{k,t} = \gamma_{k,t-1} y_{k,t}$
  (Updating)
  **for** $k = 1 \to K$ **do**
    $\hat{\mathbf{C}}_{k,t-1} = E[\boldsymbol{x}_{k,t-1}\boldsymbol{x}^{\mathsf{H}}_{k,t-1}]$
  **end for**
  **for** Iter $= 1 \to$ Maxiter **do**
    **for** $k = 1 \to K$ **do**
      $\sigma_{k,t-1} = \sqrt{\boldsymbol{w}^{\mathsf{H}}_{k,t-1}\hat{\mathbf{C}}_{k,t-1}\boldsymbol{w}_{k,t-1}}$
    **end for**
    $r_{t-1} = \sqrt{\sum_k |\boldsymbol{w}^{\mathsf{H}}_{k,t-1}\boldsymbol{x}_{k,t-1}|^2}$
    **for** $k = 1 \to K$ **do**
      $\boldsymbol{a}_{k,t-1} = \frac{\hat{\mathbf{C}}_{k,t-1}\boldsymbol{w}_{k,t-1}}{\boldsymbol{w}^{\mathsf{H}}_{k,t-1}\hat{\mathbf{C}}_{k,t-1}\boldsymbol{w}_{k,t-1}}$
      $\boldsymbol{V}_{k,t-1} = E[\boldsymbol{x}_{k,t-1}\boldsymbol{x}^{\mathsf{H}}_{k,t-1}/r_{t-1}]$
      $\boldsymbol{P}_{k,t-1} = \alpha\boldsymbol{P}_{k,t-2} + (1-\alpha)\frac{\boldsymbol{V}_{k,t-1}}{\sigma^2_{k,t-1}}$
      $\boldsymbol{a}_{\text{tmp}} = \frac{\boldsymbol{w}^{\mathsf{H}}_{k,t-1}\boldsymbol{V}_{k,t-1}\boldsymbol{w}_{k,t-1}}{\sigma^2_{k,t-1}}\boldsymbol{a}_{k,t-1}$
      $\boldsymbol{Q}_{k,t-1} = \alpha\boldsymbol{Q}_{k,t-2} + (1-\alpha)\boldsymbol{a}_{\text{tmp}}$
      $\boldsymbol{R}_{k,t-1} = \alpha\boldsymbol{R}_{k,t-2} + (1-\alpha)\boldsymbol{V}_{k,t-1}$
      $\boldsymbol{w}_{k,t-1} = \boldsymbol{P}^{-1}_{k,t-1}\boldsymbol{Q}_{k,t-1}$
      $\boldsymbol{w}_{k,t-1} = \frac{\boldsymbol{w}_{k,t-1}}{\sqrt{\boldsymbol{w}^{\mathsf{H}}_{k,t-1}\boldsymbol{R}_{k,t-1}\boldsymbol{w}_{k,t-1}}}$
    **end for**
  **end for**
**end for**

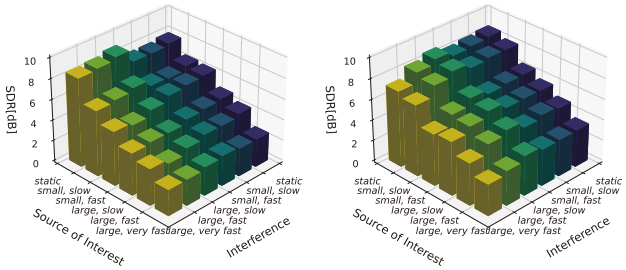
## 4. Experiment

We verified the effectiveness of online-CSV-AuxIVE for sources at different ranges and speeds compared with the conventional online-AuxIVA. In this experiment, we set the forgetting factor $\alpha = 0.3$ and the block length $L_b = 150$ for online-CSV-AuxIVE. For online-AuxIVA, we set the time-window length to 20 frames and the forgetting factor to 0.95.

### 4.1 Experimental conditions

The dataset is taken from Experiment 4.2 in [9]. The authors attached wireless speakers to a manually driven rotating arm, manually controlled the range and speed of movement of the two sources, and recorded the mixed signals of the two source signals in different states using four linear microphones spaced 16 cm apart. The male voice represents the SOI and the female voice represents the interference (IR). The layout of the experimental environment is shown in Fig. 1, with two source signals moving on a semicircular trajectory with a radius of 1 m. From Table 1, it can be seen that each source signal has six different conditions, and after combining with each other, there are 36 different mixed signals.

The signal-to-distortion ratio (SDR) of segmentation followed by averaging is used as a criterion for judging the performance of the algorithm. The SDR is calculated using BSS-eval [11]. The length of the segments is 1 s. The experimental conditions are shown in Table 2.




(a) online-AuxIVA (b) online-CSV-AuxIVE

Figure 2: Average SDRs in 3D histogram

Table 2: Experimental conditions

| | |
|---|---|
| Sampling rate | 16 kHz |
| STFT window function | Hanning |
| STFT window length | 1,024 samples |
| Number of iterations (Maxiter) | 100 |
| Initial value of $\boldsymbol{Q}$ | Zero matrix |
| Initial value of $\boldsymbol{P}$ | $10^{-12}\times$ Identity matrix |
| Room dimensions | $12 \times 8 \times 2.6\ m$ |

### 4.2 Results

We compare the extraction performance between online-AuxIVA and online-CSV-AuxIVE. As shown in Fig. 2(a), online-AuxIVA performs well only when the SOI is stationary. When the SOI starts to move, its performance decreases.

On the other hand, in Fig. 2(b), online-CSV-AuxIVE is shown to be less sensitive to the movement of the SOI. If the SOI moves slowly, online-CSV-AuxIVE performs better than online-AuxIVA, regardless of the movement condition of the IR. As shown in Fig. 3, focusing only on the condition of the SOI movement of {small, slow}, online-CSV-AuxIVE had about 1.5 dB higher SDR than online-AuxIVA regardless of the variations of the moving range and the moving speed of the interference.

### 4.3 Realistic implementation on PC

For the 36 mixed signals in the experimental dataset, the computation times of online-AuxIVA and online-CSV-AuxIVE with the parameters shown in Table 2 are 4.9 s and 3.5 s, respectively, using Python in Pycharm on a Macbook pro computer with an apple M2 chip (8-core CPU). In comparison, the proposed algorithm improves computational efficiency for the extraction of specific moving source signals.

### 4.4 Limitations in practical application

There is some space to improve the proposed algorithm for practical applications. In the experiment, the block length

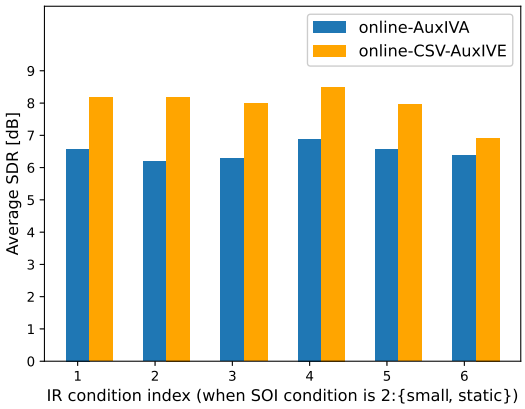



Figure 3: Average SDRs of the online-CSV-AuxIVE and online-AuxIVA algorithms (condition of SOI: {small, slow})

$L_b$ and forgetting factor $\alpha$ significantly affect the results. However, the values of these two parameters that achieve the best performance for different datasets are often different. In other words, the combination of parameters used in Sect. 4, $L_b = 150$ frames and $\alpha = 0.3$, does not necessarily give the best performance on other datasets. Therefore, in practice, we need to think about how to set the parameters for different datasets to obtain the best extraction performance.

## 5. Conclusions

In this paper, for the moving BSE, we developed a novel and effective offline CSV mixing model. To make this algorithm applicable to devices requiring real-time processing, we proposed an online version of the CSV-AuxIVE algorithm. Unlike the offline algorithm, the proposed algorithm needs to know only part of the mixed signal to process the input observed signals sequentially. The results confirmed that the average SDR of the online-CSV-AuxIVE was about 1.5 dB higher than that of the online-AuxIVA under the same conditions of the target source, regardless of the variations of the moving range and the speed of the interference.

## References

[1] A. Hyvärinen and E. Oja: Independent component analysis: Algorithms and applications, Neural Networks, Vol. 13, Nos. 4-5, pp. 411-430, 2000.

[2] A. Tharwat: Independent component analysis: An introduction, Applied Computing and Informatics, Vol. 17, No. 2, pp. 222-249, 2021.

[3] T. Kim, T. Eltoft and T.-W. Lee: Independent vector analysis: An extension of ICA to multivariate compo-

nents, Independent Component Analysis and Blind Signal Separation, pp. 165-172, 2006.

[4] T. Kim, I. Lee and T.-W. Lee: Independent vector analysis: Definition and algorithms, Fortieth Asilomar Conference on Signals, Systems and Computers, pp. 1393-1396, 2006.

[5] N. Ono: Stable and fast update rules for independent vector analysis based on auxiliary function technique, Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA), pp. 189-192, 2011.

[6] N. Ono: Auxiliary-function-based independent vector analysis with power of vector-norm type weighting functions, Proceedings of the 2012 Asia Pacific Signal and Inf. Process. Assoc. Annu. Summit and Conf. (APSIPA ASC), pp. 1-4, 2012.

[7] T. Taniguchi, N. Ono, A. Kawamura and S. Sagayama: An auxiliary-function approach to online independent vector analysis for real-time blind source separation, Joint Workshop on Hands-free Speech Communication and Microphone Arrays (HSCMA), pp. 107-111, 2014.

[8] Z. Koldovský, J. Málek and J. Janský: Extraction of independent vector component from underdetermined mixtures through block-wise determined modeling, 2019 IEEE Int. Conf. on Acoust. Speech and Signal Process. (ICASSP), pp. 7903-7907, 2019.

[9] J. Janský, Z. Koldovský, J. Málek and T. Kounovský: Auxiliary function-based algorithm for blind extraction of a moving speaker, EURASIP Journal on Audio, Speech, and Music Processing (JASMP), pp. 1-16, 2022.

[10] Z. Koldovský and P. Tichavský: Gradient algorithms for complex non-gaussian independent component/vector extraction, question of convergence. IEEE Trans. on Signal Processing, Vol. 67, No. 4, pp. 1050-1064, 2018.

[11] E. Vincent, R. Gribonval and C. Févotte: Performance measurement in blind audio source separation, IEEE Trans. on Audio, Speech, and Language Processing (ASLP), Vol. 14, No. 4, pp. 1462-1469, 2006.