Component-Dependent Independent Component Analysis for Time-Sensitive Applications

Huanzhuo Wu*, Yunbin Shen*, Jiajing Zhang*†, Ievgenii A. Tsokalo*, Hani Salah* and Frank H.P. Fitzek*†
*Deutsche Telekom Chair of Communication Networks - Technische Universität Dresden, Germany
†Centre for Tactile Internet with Human-in-the-Loop (CeTI)
Email:{*huanzhuo.wu|jiajing.zhang|ievgenii.tsokalo|hani.salah|frank.fitzek*}*@tu-dresden.de, yunbin.shen@mailbox.tu-dresden.de*

***Abstract*—In time-sensitive applications within industry 4.0, e.g. anomaly detection and human-in-the-loop, the data generated by multiple sources should be quickly separated to give the applications more time to make decisions and ultimately improve production performance. In this paper, we propose a Component-dependent Independent Component Analysis (CdICA) method that can separate multiple randomly mixed signals into independent source signals faster, for further data analysis in time-sensitive applications. Based on the Independent Component Analysis (ICA) algorithm, we first generate an initial separation matrix relying on the known mixture components, so that the separation speed of the traditional ICA can be increased. Our simulative results show that the CdICA method reduces the separation time by 55% to 83% compared to the most notable related work called FastICA and meanwhile it does not diminish the accuracy of the separation.**

***Index Terms*—Blind source separation, Time-sensitive application, IIoT, Industry 4.0**

I. Introduction

With the rise of internet technologies, the concept of Industry 4.0 was proposed, which intends to connect machines, human and other elements in the industrial field into an integrated cyber-physical system [1]. The industrial IoT (IIoT) is considered as a key supporting asset. The physical world can be represented by the digital signals, which are utilized for communication, analysis and further decisions making [2]. To ensure success in this endeavour, on one side, the signal of each element is essential for subsequent analysis and decision making. In fact, there is always more than one working element that simultaneously generates signals in the industrial field. On the other side, the time consumption of signal gathering is often a very critical factor.

For instance, in anomaly detection applications [3], the source signals produced by running machines are collected by IIoT devices and compared with those obtained under the expected working situation. Once an obvious deviation occurs, the system ought to detect the anomaly in time, so that the fault can be cleared at an early stage and the production breakdown is prevented. However, the observed signals are inevitably mixed among multiple source signals generated by each machine, while the desired result is to locate the malfunction element as accurate and fast as possible based on the collected data. Another application is human-in-the-loop. Electroencephalogram (EEG) as one of the psychological insights [4] is transferred from the human learning process to machines. The extraction of the essential features of the EEG signals in the presence of artifacts would be a problem. The machines are also expected to obtain these features under low latency conditions in order to interact timely and seamlessly with humans. These applications not only need to separate the collected mixed signals on IIoT devices, but also are sensitive to the time consumption of source separation.

A well-known linear transformation method Independent Component Analysis (ICA) [5] was developed to find a linear representation so that the mixed signals are statistically separated [6], [7]. Nevertheless, ICA is considered a time-consuming method as it separates the mixed components using Newton's Iteration. During the last few years, researchers have explored various ways to accelerate the ICA algorithm so that time-sensitive applications benefit from it. The most remarkable currently used algorithm is FastICA [8], which computes and returns independent components one after the other using new contrast functions in ICA. However, it is not faster enough to perform a complete separation. The authors of [9] proposed to apply ICA either in a sliding-window or in a cumulative mode, so that computational time becomes compatible with real-time settings for task-related signals. However, detecting and separating non-task-related signals are not supported in [9]. FastICA was implemented efficiently on customized hardware platforms such as DSP processors and FPGA [10]. Most IoT devices unfortunately do not have such special hardwares. Hence, industrial time-sensitive applications require a fast separation algorithm that can analyze various types of signals and support IIoT devices.

In this paper, a novel Component-dependent Independent Component Analysis (CdICA) method, by harnessing the property of known mixtures, is proposed. Unlike the aforementioned ICA approaches, in which the initial input for starting Newton's Iteration is a random value [11], and thus ignore the characteristics of the known mixtures, CdICA analyzes the correlation between the known mixtures and the mixing method. With our proposed CdICA, the number of Newton's Iteration can be significantly reduced to separate all source components, which we show through extensive numerical simulations. Depending on the correlation, an approx. estimation of the mixing method can be constructed as the initial input of Newton's Iteration to reduce the time consumption of component separation for time-sensitive applications.

The rest of the paper is structured as follow. First, Section II gives an overview of some basic concepts and provides a description of the problem. The proposed CdICA is then described in detail in Section III. Section IV covers the numerical evaluation and result discussion. Finally, Section V concludes the contribution and points out further perspectives.

II. Problem Statement

Following the literature, we represent the signals as column vectors, and use the following notation:

- $\mathbf{M}$: A matrix.
- $\mathbf{m}_i$: The i-th column of the matrix $\mathbf{M}$, the transpose $\mathbf{m}_i^T$ is the i-th row.
- $m_{i,j}$: The element of the matrix $\mathbf{M}$ in the i-th row in the j-th column.

A. Blind Source Separation

We use a computational method for separating a multivariate signal from observers $\mathbf{X} \in \mathbb{R}^{k \times m}$ into additive n components $\mathbf{S} \in \mathbb{R}^{n \times m}$. This method is called Blind Source Separation (BSS) [12]. The additive n components $\mathbf{S}$ are original sources in m time slots. These n components of original sources are (i) statistically independent and (ii) non-Gaussian distributed. The multivariate signals $\mathbf{X}$ are the mixtures obtained from k observers in the same m time slots.

The component $\mathbf{x}_i$ of the observed mixtures $\mathbf{X}$ at each time instant is generated as a linear combination of the source components $\mathbf{s}_i$:

$$x_{j,i} = a_{j,1}s_{1,i} + a_{j,2}s_{2,i} + \cdots + a_{j,n}s_{n,i} + e \tag{1}$$

weighted by the mixing weights $a_{j,l}$ for $j = 1, \cdots, k$ and $l = 1, \cdots, n$. When studying BSS, noise e is generally not considered. It can be assumed either that there is no noise or that noise has been negligibly. In this paper, we only consider the case in which there is no noise ($e = 0$). The generative model can be written as

$$\mathbf{x}_i = \mathbf{A}\mathbf{s}_i \quad \text{or} \quad \mathbf{X} = \mathbf{A}\mathbf{S}, \tag{2}$$

where $\mathbf{A}$ is the mixing matrix. For enabling BSS, the number of observers k should be at least as large as the number of sources n (i.e. $k \geqslant n$). Otherwise, a mixing matrix $\mathbf{A}$ will not have an inverse. In the following sections of this paper, following [8], it is assumed that the mixing matrix is a square matrix, i.e. $k = n$.

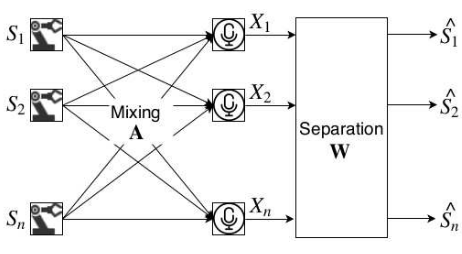

Fig. 1. Blind Source Separation Model [13]

The task of the given BSS model in Fig. 1 is to estimate both the mixing matrix $\mathbf{A}$ and the original sources $\mathbf{S}$. The estimated source $\hat{\mathbf{S}}$ can be recovered by multiplying the observed mixtures with the separation matrix:

$$\hat{\mathbf{S}} = \mathbf{A}^{-1}\mathbf{X} = \mathbf{W}\mathbf{X}. \tag{3}$$

Independent Component Analysis (ICA) [5] is one of the well-known methods for separation. It aims to be as statistically independent as possible in producing components by minimizing both second-order and higher-order dependencies in the given data. In 1999, Hyvärinen and Oja introduced a fast algorithm for ICA, known as FastICA [8].

B. Newton's Iteration of ICA

Like most algorithms of ICA, maximizing the non-Gaussianity of $\mathbf{WX}$ in Eq. (3) will give the independent components. In Eq. (4), negentropy is used as a measure of non-Gaussianity [11]. Correspondingly, FastICA seeks for such $\mathbf{W}$ so that the negentropy is maximized:

$$F(\mathbf{W}) = E\{\mathbf{X}g(\mathbf{W}^T\mathbf{X})\} - \beta\mathbf{W}. \tag{4}$$

After the given mixture data $\mathbf{X}$ is preprocessed by centering and whitening, the problem of maximizing the non-Gaussianity can be solved as an approximation of Newton's Iteration [11]:

$$\mathbf{W}_{n+1} = \mathbf{W}_n - \frac{E\{\mathbf{X}g(\mathbf{W}_n^T\mathbf{X})\} - \beta\mathbf{W}_n}{E\{g'(\mathbf{W}_n^T\mathbf{X})\} - \beta}. \tag{5}$$

This equation reveals the ways to optimize the performance of the ICA. It can be improved either by selecting a novel contrast function g or by constructing a proper initial separation matrix $\mathbf{W}_0$, so that the convergence of Newton's Iteration in ICA can be achieved faster.

In [8], the authors showed how FastICA chooses contrast functions that are robust and/or have minimum variance, so that the convergence can be very fast (cubic or at least quadratic). Therefore, we see a potential improvement in the second way of optimization to accelerate ICA.

The initial separation matrix of Newton's Iteration in ICA is nowadays filled with random values [11]. In this way, the absence of correlation between $\mathbf{W}_0$ and $\mathbf{X}$ is evident. It may lead to numerous iterations. Therefore, in Section III we will analyze from the perspective of initial separation matrix $\mathbf{W}_0$ and propose a new method to build the initial separation matrix depending on the known components $\mathbf{X}$, which could reduce the number of iterations in order to speed up the convergence and separation.

III. Our Solution: CdICA

In this section, we describe our novel Component-dependent ICA (CdICA) method that shows how an initial matrix is constructed depending on the known components. A mathematical analysis of CdICA will also be presented.

A. Dependency of Known Components and Mixing Matrix

According to Eq. (2), the expected value of the i-th mixture component $\mathbf{x}_i^T$ in m time slots can be described by Eq. (6):

$$E(\mathbf{x}_i^T) = a_{i,1}E(\mathbf{s}_1^T) + a_{i,2}E(\mathbf{s}_2^T) + \cdots + a_{i,n}E(\mathbf{s}_n^T). \quad (6)$$

Therefore, the ratio of two mixture components' expected values can be resulted from Eq. (7):

$$\frac{E(\mathbf{x}_i^T)}{E(\mathbf{x}_j^T)} = \frac{a_{i,1}E(\mathbf{s}_1^T) + a_{i,2}E(\mathbf{s}_2^T) + \cdots + a_{i,n}E(\mathbf{s}_n^T)}{a_{j,1}E(\mathbf{s}_1^T) + a_{j,2}E(\mathbf{s}_2^T) + \cdots + a_{j,n}E(\mathbf{s}_n^T)}. \quad (7)$$

In Eq. (7), it can be found that as long as $E(\mathbf{s}_l^T)$ satisfies the constraint in Eq. (8), we can get the ratio of all elements $a_{i,l}$ in the mixing matrix $\mathbf{A}$:

$$\frac{a_{i,l}}{a_{j,l}} \approx \frac{E(\mathbf{x}_i^T)}{E(\mathbf{x}_j^T)}\; s.t.\; |E(\mathbf{s}_l^T)| \gg |E(\mathbf{s}_{other}^T)|,\; l = 1, \cdots, n. \quad (8)$$

Since the original sources $\mathbf{S}$ are unknown, we select a part of the known mixture components $\mathbf{x}_i^T$ and construct them as $\widehat{\mathbf{x}_i}^T$. The $\widehat{\mathbf{x}_i}^T$ are to be generated by $\mathbf{S}$ that meet the constraint in Eq. (8). Referring to Eq. (8) and $\widehat{\mathbf{x}_i}^T$, an approximation of the mixing matrix $\hat{\mathbf{A}}$ is obtained with respect to the known mixtures $\mathbf{X}$:

$$\hat{\mathbf{A}} = \begin{bmatrix} \frac{a_{1,1}}{a_{1,1}} & \cdots & \frac{a_{1,n}}{a_{n,n}} \\ \frac{a_{2,1}}{a_{1,1}} & \cdots & \frac{a_{2,n}}{a_{n,n}} \\ \ddots & \ddots & \ddots \\ \frac{a_{n,1}}{a_{1,1}} & \cdots & \frac{a_{n,n}}{a_{n,n}} \end{bmatrix} = \begin{bmatrix} 1 & \cdots & \widehat{a_{1,n}} \\ \widehat{a_{2,1}} & \cdots & \widehat{a_{2,n}} \\ \ddots & \ddots & \ddots \\ \widehat{a_{n,1}} & \cdots & 1 \end{bmatrix}. \quad (9)$$

In a further step, an inverse matrix:

$$\hat{\mathbf{W}} = \hat{\mathbf{A}}^{-1} \quad (10)$$

can be derived. Comparing with a random matrix in FastICA, the estimated matrix $\hat{\mathbf{W}}$ by CdICA should have a higher similarity to the separation matrix $\mathbf{W}$ in Eq. (3). The similarity of these two matrices will be evaluated in Section IV-B1.

Hence, the matrix $\hat{\mathbf{W}}$ in Eq. (10) can be used as the initial matrix $\hat{\mathbf{W}}_0$ for Newton's Iteration in Eq. (5), in which case the convergence of Newton's Iteration is accelerated by reducing the number of iterations, because the estimated initial matrix $\hat{\mathbf{W}}_0$ has higher similarity to the separation matrix $\mathbf{W}$. Section IV-B2 and Section IV-B3 will numerically evaluate the influence of $\hat{\mathbf{W}}_0$ on the number of iterations and the time consumption of separation. Moreover, the proposed CdICA method has no effect on the separation quality, since changing the initial matrix only affects the speed of Newton's Iteration but its convergence remains unchanged, which means that the final separation matrix stays the same. In Section IV-B4 the separation quality of CdICA will be evaluated.

B. Construction of Initial Separation Matrix

In this section, we take a look at how to select $\widehat{\mathbf{x}_i}^T$ from known mixture components, so that $\widehat{\mathbf{x}_i}^T$ is derived from the $\mathbf{s}_l{}^T$ following the Eq. (8), to construct the initial separation matrix $\hat{\mathbf{W}}_0$.

The mixing matrix $\mathbf{A}$ can be scaled and shifted to match the elements of $\hat{\mathbf{A}}$ in Eq. (9), i.e. the diagonal element is 1, while the non-diagonal elements are between 0 and 1:

$$\begin{cases} a_{i,j} = 1 & i = j \\ 0 < a_{i,j} < 1 & i \neq j. \end{cases} \quad (11)$$

The results of scaling and shifting $\mathbf{A}$ are the permutation of estimated sources' order and the scaling of estimated sources' amplitude. These two are the ambiguities of the ICA algorithm itself, which have been discussed in [11].

Starting with analyzing the number of mixture components $n = 2$, we can then extend n to arbitrary values. In the following, we will first prove that $|E(\mathbf{s}_1{}^T)|$ in the selected $\widehat{\mathbf{x}_i}^T$ is greater than 0, and we then show that $|E(\mathbf{s}_2{}^T)|$ is smaller than $|E(\mathbf{s}_1{}^T)|$ and close to or equal to 0.

According to Eq. (2), the mixtures in Eq. (12) can be represented with Eq. (13):

$$\begin{bmatrix} \mathbf{x}_1{}^T \\ \mathbf{x}_2{}^T \end{bmatrix} = \begin{bmatrix} 1 & a_{1,2} \\ a_{2,1} & 1 \end{bmatrix} \begin{bmatrix} \mathbf{s}_1{}^T \\ \mathbf{s}_2{}^T \end{bmatrix}, \quad (12)$$

$$\begin{aligned} f_1(\mathbf{s}_1{}^T) &= \mathbf{s}_1{}^T + a_{1,2}\mathbf{s}_2{}^T \\ f_2(\mathbf{s}_1{}^T) &= a_{2,1}\mathbf{s}_1{}^T + \mathbf{s}_2{}^T. \end{aligned} \quad (13)$$

When $f_1(\mathbf{s}_1{}^T) > f_2(\mathbf{s}_1{}^T)$, it is easily to find that:

$$\mathbf{s}_1{}^T > \frac{1 - a_{1,2}}{1 - a_{2,1}}\mathbf{s}_2{}^T \geqslant 0,\; \mathbf{s}_2{}^T \geqslant 0, \quad (14)$$

and when $f_1(\mathbf{s}_1{}^T) > -f_2(\mathbf{s}_1{}^T)$,

$$\mathbf{s}_1{}^T > -\frac{1 + a_{1,2}}{1 + a_{2,1}}\mathbf{s}_2{}^T > 0,\; \mathbf{s}_2{}^T < 0, \quad (15)$$

as illustrated in Fig. 2.

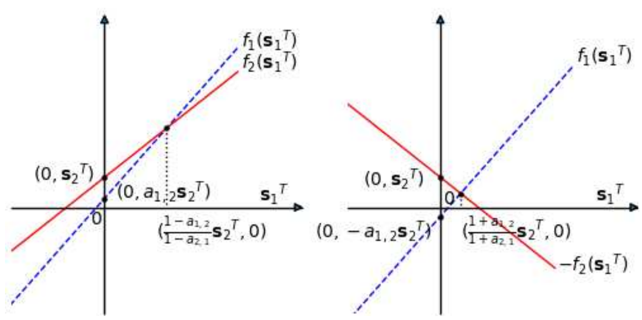

Fig. 2. $\widehat{\mathbf{s}_1}^T$ and $\widehat{\mathbf{x}_1}^T$ $\widehat{\mathbf{x}_2}^T$ when $\mathbf{s}_2{}^T \geqslant 0$ and $\mathbf{s}_2{}^T < 0$

The known mixture component that meets both of the above conditions is selected to build the initial matrix $\hat{\mathbf{W}}_0$. They are noted as $\widehat{\mathbf{x}_1}^T$ and $\widehat{\mathbf{x}_2}^T$. The unknown source components, which generate $\widehat{\mathbf{x}_1}^T$ and $\widehat{\mathbf{x}_2}^T$, are noted as $\widehat{\mathbf{s}_1}^T$ and $\widehat{\mathbf{s}_2}^T$. Eq. (14) and Eq. (15) are equivalent to say that for any value of $\widehat{\mathbf{s}_2}^T$:

$$\widehat{\mathbf{s}_1}^T > \xi > 0\; s.t.\; \widehat{\mathbf{x}_1}^T > |\widehat{\mathbf{x}_2}^T|, \quad (16)$$

it can be proved that

$$|E(\widehat{\mathbf{s}_1}^T)| = E(\widehat{\mathbf{s}_1}^T) > 0\; s.t.\; \widehat{\mathbf{x}_1}^T > |\widehat{\mathbf{x}_2}^T|. \quad (17)$$

After selecting $\widehat{\mathbf{x}_1}^T$ that fulfills Eq. (17), we select a part of $\widehat{\mathbf{x}_1}^T$, so that $|E(\widehat{\mathbf{s}_1}^T)| \gg |E(\widehat{\mathbf{s}_2}^T)|$.

From Eq. (13) and Eq. (16), it is easy to see that

$$\begin{aligned} f_1(\widehat{\mathbf{s}_1}^T) + f_2(\widehat{\mathbf{s}_1}^T) > 0 \\ f_1(\widehat{\mathbf{s}_1}^T) - f_2(\widehat{\mathbf{s}_1}^T) > 0. \end{aligned} \tag{18}$$

Hence,

$$\begin{aligned} \widehat{\mathbf{s}_1}^T > f_3(\widehat{\mathbf{s}_2}^T) = -\frac{1+a_{1,2}}{1+a_{2,1}}\widehat{\mathbf{s}_2}^T \geqslant 0 \\ \widehat{\mathbf{s}_1}^T > f_4(\widehat{\mathbf{s}_2}^T) = \frac{1-a_{1,2}}{1-a_{2,1}}\widehat{\mathbf{s}_2}^T \geqslant 0. \end{aligned} \tag{19}$$

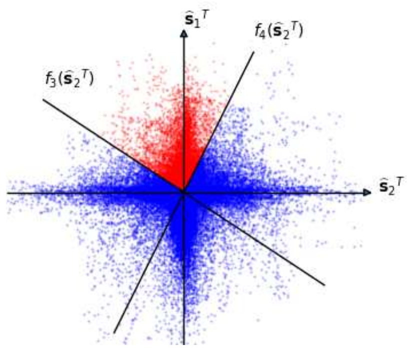

Fig. 3. Joint Distribution of $\widehat{\mathbf{s}_1}^T$ and $\widehat{\mathbf{s}_2}^T$

Fig. 3 shows the joint distribution of $\widehat{\mathbf{s}_1}^T$ and $\widehat{\mathbf{s}_2}^T$, the values marked as red are constrained by Eq. (19). For an arbitrary number of sources, the same principle is applied. Together with Eqs. (11) and (19), the boundary of the joint distribution of $\widehat{\mathbf{s}_1}^T$ and $\widehat{\mathbf{s}_2}^T$ can be obtained in Eq. (20):

$$\begin{aligned} -2 < \frac{\partial f_3(\widehat{\mathbf{s}_2}^T)}{\partial \widehat{\mathbf{s}_2}^T} < -\frac{1}{2} \\ 0 < \frac{\partial f_4(\widehat{\mathbf{s}_2}^T)}{\partial \widehat{\mathbf{s}_2}^T} < \infty. \end{aligned} \tag{20}$$

According to Eq. (20), we note that there are always a group of negative values of $\widehat{\mathbf{s}_2}^T$ selected above $f_3(\widehat{\mathbf{s}_2}^T)$. Depending on the value of $a_{2,1}$, a group of positive values of $\widehat{\mathbf{s}_2}^T$ are selected above $f_4(\widehat{\mathbf{s}_2}^T)$. Above $f_3(\widehat{\mathbf{s}_2}^T)$ and $f_4(\widehat{\mathbf{s}_2}^T)$, the values of $\widehat{\mathbf{s}_1}^T$ are always positive:

$$\begin{cases} \exists \hat{s}_{2,i} \in \widehat{\mathbf{s}_2}^T : \hat{s}_{2,i} \leqslant 0 \\ \exists \hat{s}_{2,i} \in \widehat{\mathbf{s}_2}^T : \hat{s}_{2,i} \geqslant 0 \\ \forall \hat{s}_{1,i} \in \widehat{\mathbf{s}_1}^T : \hat{s}_{1,i} \geqslant 0. \end{cases} \tag{21}$$

Eq. (21) clearly shows that $|E(\widehat{\mathbf{s}_1}^T)| > |E(\widehat{\mathbf{s}_2}^T)|$. When $a_{1,2} = a_{2,1} \neq 1$ and $\mathbf{X}$ is centered, the selected $\widehat{\mathbf{x}_i}^T$ leads to $|E(\widehat{\mathbf{s}_1}^T)| \gg |E(\widehat{\mathbf{s}_2}^T)| = 0$, which means that the separation matrix can already be estimated. An extreme situation is $a_{2,1} \approx 1$, in which there will be an exception $|E(\widehat{\mathbf{s}_1}^T)| \approx |E(\widehat{\mathbf{s}_2}^T)|$. However, the possibility of $a_{2,1} \approx 1$ is very low (see Eq. (9)).

Generally speaking, one can prove that as long as a subset of the known mixture components $\hat{\mathbf{X}} \subseteq \mathbf{X}$ are selected under such condition $\mathbf{x}_l^T > |\mathbf{x}_{other}^T|$, the source components $\hat{\mathbf{S}}$ which generate $\hat{\mathbf{X}}$ can satisfy the constraint $|E(\widehat{\mathbf{s}_l}^T)| \gg |E(\widehat{\mathbf{s}_{other}}^T)|$ in the Eq. (8). Therefore, Eq. (8) can be rewritten to Eq. (22), in order to construct an initial matrix $\hat{\mathbf{W}}_0 = \hat{\mathbf{A}}^{-1}$ depending on the known mixture components $\mathbf{X}$.

$$\begin{aligned} \widehat{a_{i,l}} = \frac{a_{i,l}}{a_{l,l}} \approx \frac{E(\widehat{\mathbf{x}_i}^T)}{E(\widehat{\mathbf{x}_l}^T)},\ l = 1, \cdots, n \\ \hat{\mathbf{X}} \subseteq \mathbf{X}\ s.t.\ \mathbf{x}_l^T > |\mathbf{x}_{other}^T|,\ l = 1, \cdots, n, \end{aligned} \tag{22}$$

IV. Evaluation

The CdICA introduced in the previous section is going to be evaluated from the perspectives of the initial matrix construction, the iteration number and time consumption, the separation quality. We compare CdICA with FastICA [8], which is the current most fast and robust ICA algorithm by choosing a proper constrain function to speedup ICA.

A. Setup and Testing Data

We picked random positive numbers following the normal distribution $\mathcal{N}(\mu = 0, 0 < \sigma^2 < 0.11)$ as the non-diagonal elements of the mixing matrix $\mathbf{A}$; the value of the diagonal elements was set to 1. The elements constructing the mixing matrix $\mathbf{A}$ were not only randomly distributed, but also covered 99.73% possible values under three standard deviations. Therefore, a metric Λ can be determined to represent the deviation of the randomly generated mixing matrix:

$$\Lambda = \frac{\sigma}{0.33} \times 100\%. \tag{23}$$

$n = 8, 12, 16$ audio signals were randomly selected from 63 human voices as example, which are regarded as the original sources $\mathbf{S}$. More types of data in other IoT domains will be tested in the future. Each of them was a 4-second audio signal with $16000Hz$ sampling rate. The length of each source was $m = 64000$. The mixtures $\mathbf{X}$ were generated by Eq. (2), as the known input of the CdICA and FastICA.

The CdICA was applied to construct initial matrices $\hat{\mathbf{W}}_0$ from $n = 8, 12, 16$ observed mixtures, according to Eq. (22). For comparison purposes, random matrices $\mathbf{W}_0$ were chosen as initial matrices for FastICA. For each given mixing matrix $\mathbf{A}$, 30 samples were tested. A confidence interval with 99% confidence level was also given. The Newton's Iteration was implemented by using *scikit-learn* [14].

B. Results

1) Difference from Separation Matrix: Cosine distance Δ in the Eq. (24) was used to indicate the similarity of two matrices [15]. The cosine distance is advantageous because even if the two similar matrices are far apart by the Euclidean distance, they could still have a smaller orientation. Using different mixing matrices $\mathbf{A}$, the initial matrices $\hat{\mathbf{W}}_0$ of CdICA and $\mathbf{W}_0$ of FastICA were built, their consine distances Δ to the separation matrices $\mathbf{W} = \mathbf{A}^{-1}$ are shown in Fig. 4.

$$\Delta = 1 - \frac{\mathbf{W}_{initial} \cdot \mathbf{W}}{\|\mathbf{W}_{initial}\|_2 \|\mathbf{W}\|_2}. \tag{24}$$

It can be clearly seen that by a given number of sources, the distance between our initial matrix $\hat{\mathbf{W}}_0$ and the separation matrix $\mathbf{W}$ is much smaller than the distance between the random matrix $\mathbf{W}_0$ and $\mathbf{W}$. This is because, the elements of the initial matrix $\hat{\mathbf{W}}_0$ constructed by the CdICA are obtained by the selected components $\widehat{\mathbf{x}_i}$ of the known mixtures, which shows the approximate ratio of the elements in mixing matrix $\mathbf{A}$. Conversely, the traditional FastICA method simply constructs a random initial matrix $\mathbf{W}_0$ that does not contain any information of the mixing matrix $\mathbf{A}$.

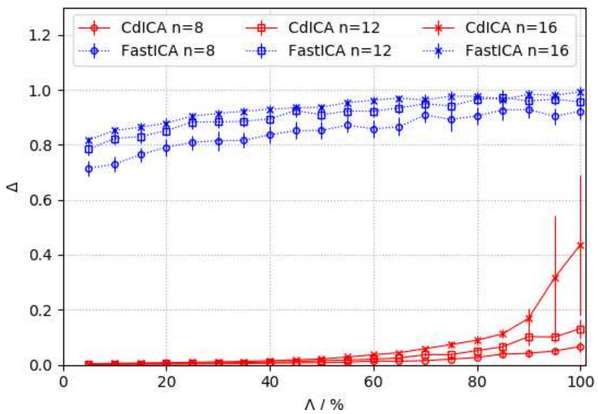

Fig. 4. Difference from Separation Matrix

2) Iteration Number: After completing the construction of initial matrices for Newton's Iteration, we investigated the effect of CdICA on the iteration number N_{inter}.

Fig. 5 shows the iteration number of CdICA and FastICA with different numbers of sources, until the convergence reached. For CdICA, about two to ten iterations were usually sufficient to estimate the separation matrix $\mathbf{W}$. Along the varying number of sources, about 14 to 23 iterations by FastICA are required until convergence is achieved. The confidence interval of FastICA is much bigger than CdICA, which means that the number of iterations required by FastICA is highly unstable.

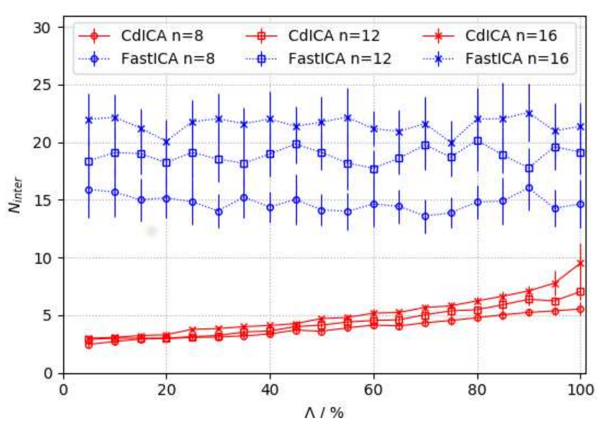

Fig. 5. Number of Newton's Iteration

The observations can be interpreted as follows. First, the initial matrix $\hat{\mathbf{W}}_0$ generated by CdICA has a higher similarity to the target separation matrix $\mathbf{W}$ (see Fig. 4), so that the number of iterations is greatly reduced. Second, the initial matrix $\hat{\mathbf{W}}_0$ of CdICA is generated based on the known mixture components $\widehat{\mathbf{x}_i}$ that already contain the mixing coefficients of the source components, as a result the number of iterations by CdICA is more stable than FastICA.

3) Time Consumption: The separation speed of CdICA was compared with the speed of FastICA. As discussed above, the convergence of Newton's Iteration needs more iterations from a random matrix than from our estimated initial separation matrix. Therefore, the time consumption of separation T with CdICA is less than with FastICA.

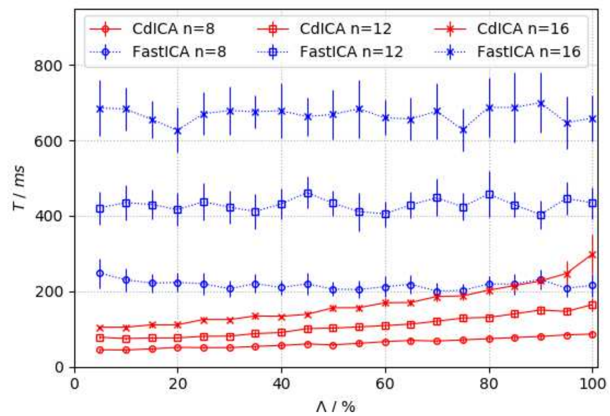

Fig. 6. Time Consumption

Fig. 6 illustrates the overall time consumption, including the time from generating the initial separation matrix of CdICA and the random matrix of FastICA until separating mixtures of CdICA and FastICA. Fig. 7 shows the ratio R_t of the time consumption with CdICA to the original time consumption with FastICA:

$$R_t = \frac{T_{CdICA}}{T_{FastICA}} \times 100\%. \tag{25}$$

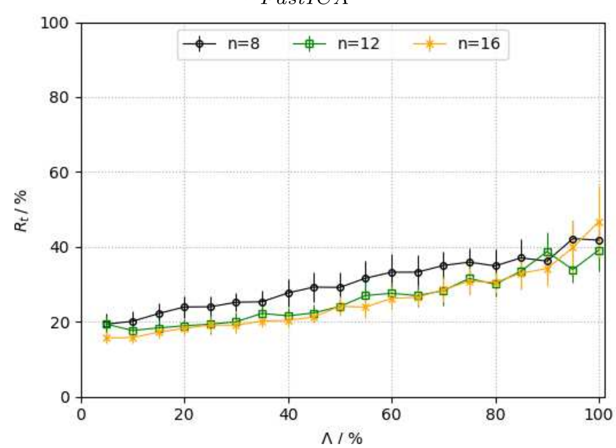

Fig. 7. Reduction of Time Consumption

It is noticeable that CdICA reduces the time required to separate the mixtures considerably by 55% to 83% compared to FastICA, by constructing an initial matrix $\hat{\mathbf{W}}_0$. Furthermore, the larger the number of sources is, the more the reduction in time. Because for a large number of sources, the time consumption of Newton's Iteration is significantly reduced to obtain a separation matrix. However, the time required to separate the mixtures through the separation matrix could normally not be significantly affected by the number of sources.

4) Separation Quality: The ratio of the original sources $\mathbf{S}$ to the difference between $\mathbf{S}$ and the estimated sources $\hat{\mathbf{S}}$, SNR, is the indicator of the separation quality:

$$SNR = \sum_{i=1}^{n} 20 \log_{10} \frac{|\mathbf{S}_i|}{|\hat{\mathbf{S}}_i - \mathbf{S}_i|}. \tag{26}$$

The separation qualities of CdICA and FastICA are shown in Fig. 8. One can notice that the simulation results of these two methods have no significant difference. CdICA has to look for an initial matrix with a shorter distance to the separation matrix $\mathbf{W}$, which can accelerate the iteration speed, but has no impact on the convergence of Newton's Iteration.

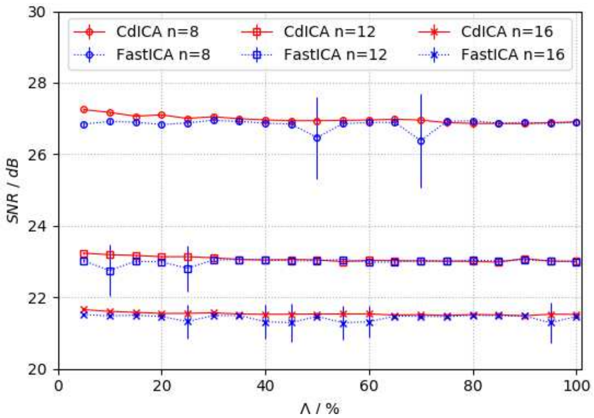

Fig. 8. Separation Quality

C. Discussion

From the experimental results mentioned above, we can first note that with the help of CdICA, the separation time in Fig. 6 and Fig. 7 is greatly reduced, and it is also stable across different mixing matrices. The reason is that the initial matrix of CdICA is estimated based on the known mixture components. The accuracy of the estimated initial matrix to the separation matrix is better than a random matrix, shown in Fig. 4. By improving the accuracy of the initial matrix, the Newton's Iteration needs less iterations to reach the convergence (Fig. 5), thus the estimation of the separation matrix is accelerated. More notable is that even the separation time is reduced, it is not at the sacrifice of the separation quality (Fig. 8).

V. Conclusion

This paper presented and evaluated a novel Component-dependent Independent Component Analysis (CdICA) method for time-sensitive applications. In CdICA, the known components are used to generate a component dependent initial separation matrix, so that the separation time is reduced by 55% to 83%. CdICA does not loss any quality, since the separation method is not changed. Being based on FastICA, CdICA is compatible with current FastICA and any contrast functions. Extensive numerical evaluations showed that with CdICA the original signals can be separated in an early stage, and thus more time can be used for time-sensitive applications to make decisions in industrial fields.

We have likewise noticed that there are more possibilities to improve the speed of the ICA algorithm, e.g. accelerating the estimation of the initial matrix, improving the accuracy of selected known components, and dynamically adjusting the iteration step. The aforementioned three approaches will be the focus of our future research.

Acknowledgment

This work has been partially supported by the Federal Ministry of Education and Research of the Federal Republic of Germany (Foerderkennzeichen 16KIS0727, 5Gang), and partially funded by the German Research Foundation (DFG, Deutsche Forschungsgemeinschaft) as part of Germany's Excellence Strategy – EXC 2050/1 – Project ID 390696704 – Cluster of Excellence "Centre for Tactile Internet with Human-in-the-Loop" (CeTI) of Technische Universität Dresden.

References

[1] M. Wollschlaeger, T. Sauter, and J. Jasperneite, "The future of industrial communication: Automation networks in the era of the internet of things and industry 4.0," *IEEE Industrial Electronics Magazine*, vol. 11, no. 1, pp. 17–27, 2017.

[2] B. Sniderman, M. Mahto, and M. J. Cotteleer, "Industry 4.0 and manufacturing ecosystems: Exploring the world of connected enterprises," *Deloitte Consulting*, 2016.

[3] S. Ludwig, M. Karrenbauer, A. Fellan, H. D. Schotten, H. Buhr, S. Seetaraman, N. Niebert, A. Bernardy, V. Seelmann, V. Stich *et al.*, "A5g architecture for the factory of the future," in *2018 IEEE 23rd International Conference on Emerging Technologies and Factory Automation (ETFA)*, vol. 1. IEEE, 2018, pp. 1409–1416.

[4] T.-P. Jung, S. Makeig, M. J. McKeown, A. J. Bell, T.-W. Lee, T. J. Sejnowski *et al.*, "Imaging brain dynamics using independent component analysis," *Proceedings of the IEEE*, vol. 89, no. 7, pp. 1107–1122, 2001.

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

[6] M. Atzmueller, D. Arnu, and A. Schmidt, "Anomaly detection and structural analysis in industrial production environments," in *Data Science–Analytics and Applications*. Springer, 2017, pp. 91–95.

[7] A. Delorme and S. Makeig, "Eeglab: an open source toolbox for analysis of single-trial eeg dynamics including independent component analysis," *Journal of neuroscience methods*, vol. 134, no. 1, pp. 9–21, 2004.

[8] A. Hyvarinen, "Fast and robust fixed-point algorithms for independent component analysis," *IEEE transactions on Neural Networks*, vol. 10, no. 3, pp. 626–634, 1999.

[9] F. Esposito, E. Seifritz, E. Formisano, R. Morrone, T. Scarabino, G. Tedeschi, S. Cirillo, R. Goebel, and F. Di Salle, "Real-time independent component analysis of fmri time-series," *Neuroimage*, vol. 20, no. 4, pp. 2209–2224, 2003.

[10] M. Turqueti, J. Saniie, and E. Oruklu, "Real-time independent component analysis implementation and applications," in *2010 17th IEEE-NPSS Real Time Conference*. IEEE, 2010, pp. 1–7.

[11] A. Hyvärinen and E. Oja, "Independent component analysis: algorithms and applications," *Neural networks*, vol. 13, no. 4-5, pp. 411–430, 2000.

[12] P. Comon and C. Jutten, *Handbook of Blind Source Separation: Independent component analysis and applications*. Academic press, 2010.

[13] H. Wu, I. A. Tsokalo, M. Taghouti, H. Salah, and F. H. P. Fitzek, "Compressible source separation in industrial iot broadband communication," in *2019 IEEE 24th International Conference on Emerging Technologies and Factory Automation (ETFA)*. IEEE, 2019.

[14] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg *et al.*, "Scikit-learn: Machine learning in python," *Journal of machine learning research*, vol. 12, no. Oct, pp. 2825–2830, 2011.

[15] B. Li and L. Han, "Distance weighted cosine similarity measure for text classification," in *International Conference on Intelligent Data Engineering and Automated Learning*. Springer, 2013, pp. 611–618.