Fuzzy Clustering of Independent Components within Time-Domain Blind Audio Source Separation Method

Jiří Málek and Zbyněk Koldovský
Faculty of Mechatronics, Informatics and Interdisciplinary Studies
Technical University in Liberec
Liberec, Czech Republic
E-mail: jiri.malek@tul.cz

***Abstract*—This paper deals with several modifications of an existing Blind Audio Source Separation (BASS) method called T-ABCD . The method applies Independent Component Analysis (ICA) in the time-domain, which gives independent components of individual signals that form unknown groups. The need is to recover these groups using a clustering algorithm and a similarity measure, and reconstruct the separated signals from the groups then. In this paper, several novel criteria that are suitable to measure the similarity between audio components are proposed. Next, fuzzy clustering algorithms are applied to group the components, and novel reconstruction approaches relying on proper weighting of components are proposed. The proposed modifications are compared by experiments, and conclusions are drawn.**

I. Introduction

The general goal of Blind Source Separation [4] is to estimate d unknown sources from a set of m mixtures observed on sensors. The estimation is performed with no prior information neither about the sources nor the mixing process. In acoustics, the mixtures are often modeled by linear convolutive mixing model

$$x_i(n) = \sum_{k=1}^{d}\sum_{\tau=0}^{M_{ik}} a_{ik}(\tau)\cdot s_k(n-\tau), \qquad (1)$$

where $x_1(n)\dots x_m(n)$ are the signals observed on microphones, and $s_1(n)\dots s_m(n)$ are the original unknown sources. The unknown parameters $a_{ik}(\tau)$ represent the source-sensor impulse responses of length M_{ik}, i.e. impulse responses expressing the propagation of sound from the position of each source to each microphone.

Assuming the mutual independence of the original sources, the sources can be retrieved using ICA [5]. However, ICA methods are usually designed to separate instantaneous mixtures ($M_{ik} = 0$), so the convolutive mixture (1) must be transformed. In this paper, the *time-domain transform* [4], [5], [6], [7] is considered, in which the convolution in (1) is expressed through the vector/matrix product. Also, due to the indeterminacy of original signal spectra, the goal is to retrieve the microphone responses (images) of sources, i.e.

$$s_k^i(n) = \sum_{\tau=0}^{M_{ik}} a_{ik}(\tau)s_k(n-\tau), \qquad (2)$$

which is the response of the kth signal on the ith microphone.

This paper addresses an existing algorithm for BASS called T-ABCD [1]. This acronym stands for Time-domain Audio source Blind separation based on the Complete Decomposition. It reflects the fact that the method applies ICA without any constraint and decomposes the whole subspace spanned by the observed signals into independent components. The method exhibits an advantageous modular structure and is thus opened to novel modifications.

The original proposal of T-ABCD from [8] does not reflect the fact that the components obtained by ICA cannot be independent in full, because the transformed mixture (1) into an istantaneous one is not truly determined [17]. Hence, the performance of the further steps of the method that reconstruct the separated sources from clustered components is limited.

Several modifications of T-ABCD were already proposed. In [3], an improvement is proposed which aims at an advanced weighting of components taking the residual interference between them into account. Similarly, [2] proposes to use a fuzzy clustering of components. This paper provides a survey of the already proposed modifications and compares them in several experiments. The comparison involves three criteria of similarity of components, two clustering algorithms and three weighting schemes.

II. Original T-ABCD

This section provides a brief description of the original T-ABCD ; for more details see [1]. The algorithm consists of four steps.

1) Definition of the observation space: As the first step, the *observation matrix* $\mathbf{X}$ whose rows span the *observation space* is defined. The rows contain delayed signals from microphones, specifically, the nth column of $\mathbf{X}$ is defined as

$$\mathbf{X}(n) = [x_1(n), x_1(n-1),\dots,x_1(n-L+1), x_2(n),\dots \quad \dots,x_m(n-L+1)]^T, \qquad (3)$$

where L is a free integer parameter. Thanks to the structure of $\mathbf{X}$, a MISO filtering of the signals from microphones can be done by multiplying $\mathbf{X}$ by a vector of length mL. Therefore,

978-1-61284-398-8/11/$26.00 ©2011 IEEE

applying an ICA algorithm to $\mathbf{X}$ can be regarded as a searching for MISO filters that output independent signals (components).

2) Decomposition of $\mathbf{X}$*:* A selected ICA algorithm[1] is applied to the observation matrix $\mathbf{X}$, which gives a *de-mixing* matrix $\mathbf{W}$. The rows of $\mathbf{W}$ correspond to separating MISO filters whose outputs $\mathbf{C} = \mathbf{W}\mathbf{X}$ are as independent as possible and are called the *independent components* (ICs).

The ICs are expected to be equal to arbitrarily filtered versions of the original signals, and their order is random. Both phenomenons are caused by the indeterminacy of ICA.

3) Similarity of ICs and their clustering: The information about which components belong to which source must therefore be retrieved. A possible way is to group the components based on their mutual similarity. In [8], the similarity is measured by projections to signal subspaces as described in Section III, where two alternative measures are introduced.

The grouping of ICs is in [8] performed via Standard Agglomerative Hierarchical Non-overlapping algorithm (SAHN) [10] with average linking strategy. It produces *hard* partitioning determined by elements of a $d \times mL$ affiliation matrix $\mathbf{U}$, which are equal either to 1 or 0 depending on whether a component belongs to a cluster or not. Nevertheless, this partitioning can be exact only if each component contain no residual interference. This suggests to use a fuzzy clustering approach, which we consider in Section IV; see also [2].

4) Reconstruction of source responses on microphones: The reconstruction aims at getting the individual microphone responses of each original source using components of the corresponding cluster. The retrieval of the source that corresponds to the kth cluster proceeds by weighting components depending on their affiliation to the kth cluster, and the weighted components are "mixed" back by the inversion matrix of $\mathbf{W}$. *Binary* weights are defined as $\mathbf{\Lambda}_{kj} = \mathbf{U}_{kj}$ provided that $\mathbf{U}$ gives a hard partitioning. In [3] it was proposed to define *fuzzy* weights according to

$$\mathbf{\Lambda}_{kj} = \left(\frac{\sum_{i \in K_k, i \neq j} \mathbf{D}_{ij}}{\sum_{i \notin K_k, i \neq j} \mathbf{D}_{ij}} \right)^{\alpha}, \tag{4}$$

where K_k contains the indices of components assigned to the kth cluster and α is a free parameter. Further definitions of the weights are proposed in Section IV.

The weighting of ICs and their re-mixing is mathematically expressed by

$$\widehat{\mathbf{S}}_k = \mathbf{W}^{-1} \text{diag}[\mathbf{\Lambda}_{k1}, \dots \mathbf{\Lambda}_{k(mL)}] \mathbf{W}\mathbf{X}. \tag{5}$$

In an ideal case, $\widehat{\mathbf{S}}_k$ is such that its nth column is equal to

$$\mathbf{S}_k(n) = [s_k^1(n), s_k^1(n-1), \dots, s_k^1(n-L+1), \\ s_k^2(n), \dots \quad \dots, s_k^m(n-L+1)]^T, \tag{6}$$

which is the contribution of the kth source to $\mathbf{X}(n)$ (see (1)-(3)). Therefore, the estimate of the response of the kth source on the ith microphones is computed according to

$$\hat{s}_k^i(n) = \frac{1}{L} \sum_{\ell=1}^{L} (\widehat{\mathbf{S}}_k)_{(i-1)L+\ell, n+\ell-1}, \tag{7}$$

where $\widehat{\mathbf{S}}_{\alpha,\beta}$ is the $\alpha\beta$th coefficient of $\widehat{\mathbf{S}}$. This concludes the algorithm.

[1]In this paper, we apply the BGSEP algorithm from [15].

III. Similarity of ICs

Let the similarity measure between the ith and jth component, $i \neq j$, be defined as the ijth element of matrix $\mathbf{D}$. In this respect, diagonal elements of $\mathbf{D}$ have no significance and can be arbitrary. We propose following three similarity measures between ICs.

A. Projections

Let $c_i(n)$ denote the nth sample of the ith component, i.e. $c_i(n) = \mathbf{C}_{in}$. Let $\mathbf{c}_i$ denote the ith row of $\mathbf{C}$. If the ith and jth component come from the same source and contain no interference, they are equal to arbitrarily filtered versions of the source, and there exist a filter f such that

$$c_i(n) = \sum_{\tau=-\infty}^{+\infty} f(\tau) c_j(n-\tau). \tag{8}$$

Therefore, the minimum mean square distance between the two sides of (8) minimized over f reflects the similarity between the components. In practice, the minimization proceeds over filters of length $2L$.

Consequently, the similarity measure is computed as

$$\mathbf{D}_{ij} = \hat{\text{E}}[\mathbf{P}_i \mathbf{c}_j]^2 + \hat{\text{E}}[\mathbf{P}_j \mathbf{c}_i]^2, \tag{9}$$

where $\mathbf{P}_i$ denotes a projector on a subspace spanned by delayed copies of the ith component, that is, by signals $c_i(n-L+1), \dots, c_i(n+L-1)$. The computation of (9) can be done efficiently using the FFT and Levinson-Durbin algorithm. This measure was introduced in [8] and used later, e.g., in [3], [1].

B. GCC-PHAT

Let $C_i(k)$ and $C_j(k)$ denote the Discrete Fourier transform of $c_i(n)$ and $c_j(n)$, respectively, and k denotes the frequency index. The GCC-PHAT coefficients of the components [9], denoted by $g_{ij}(n)$, are equal to the inverse Discrete Fourier transform of

$$G_{ij}(k) = \frac{C_i(k) \cdot C_j(k)^*}{|C_i(k)| \cdot |C_j(k)|}, \tag{10}$$

where * denotes the complex conjugation.

If the components correspond exactly to the same source up to an unknown linear-phase filter, $g_{ij}(n)$ is equal to the delayed unit impulse function, where the delay cannot be greater than L. Hence, the similarity between the ith and jth component can be measured by

$$\mathbf{D}_{ij} = \sum_{n=-L}^{L} |g_{ij}(n)| \qquad i,j = 1, \dots, mL, \, i \neq j, \tag{11}$$

C. Coherences

The coherence between two signals reflects the extend to which one signal is equal to a filtered version of the other signal, so it can be used to measure the similarity. Let $C_i(\omega_k, m)$ denote the short-time Fourier Transform of $c_i(n)$ where $m = 1, \ldots, Q$ is the index of the time-window of length P, and ω_k is the kth frequency. The coherence similarity of the ith and jth component is computed as

$$\mathbf{D}_{ij} = \frac{1}{P}\sum_{k=1}^{P} \frac{\left|\sum_{m=1}^{Q} C_i(\omega_k, m)\overline{C_j(\omega_k, m)}\right|^2}{(\sum_{m=1}^{Q} |C_i(\omega_k, m)|^2)(\sum_{m=1}^{Q} |C_j(\omega_k, m)|^2)}. \tag{12}$$

IV. Fuzzy Clustering and Weighting of ICs

A direct use of the hard partitioning performed by the SAHN algorithm consists in applying binary weights in (5). An alternative way is to re-define the weights as in (4), which can be seen as a "fuzzification" of the hard partitioning of ICs. In this section, we describe several other ways how to define the weights, in particular, when the partitioning of ICs is not hard but *fuzzy*.

A fuzzy partitioning is described by an affiliation matrix $\mathbf{U}$ whose elements satisfy

$$\begin{array}{l} \mathbf{U}_{kj} \in [0, 1] \\ \sum_{k=1}^{c} \mathbf{U}_{kj} = 1 \\ \sum_{j=1}^{K} \mathbf{U}_{kj} > 0, \end{array} \tag{13}$$

where K is the number of clustered objects (here equal to mL) and c is the assumed number of clusters.

In our task, the objects to-be clustered (the ICs) are not represented by (feature) vectors whose similarity or distance can be defined in any way. Instead, the similarity of objects is already given by matrix $\mathbf{D}$. This is called the *relational description* [11].

We examine several clustering methods working with the relational description. Namely,

- the Relational fuzzy c-medoids algorithm (RFCMdd) [12],
- the Relational possibilistic c-means algorithm (RPCM) [13], and
- the Relational fuzzy c-means algorithm (RFCM) [13].

We describe RFCM in more details in the following subsection as our experiments (Section V-A) point to its suitability within T-ABCD.

A. Relational Fuzzy C-Means Algorithm

RFCM is a relational version of the well known Fuzzy c-means algorithm (FCM) [14]. The input of FCM are K feature vectors $\mathbf{y}_1, \ldots, \mathbf{y}_K$ and the number of to-be found clusters c. FCM seeks the affiliation matrix $\mathbf{U}$ by minimizing the objective function

$$J_f(\mathbf{U}, \mathbf{G}) = \sum_{k=1}^{c}\sum_{j=1}^{K} (\mathbf{U}_{kj})^f \mathbf{V}_{kj}, \tag{14}$$

where $f > 1$ is a so-called *fuzzyfication* parameter, and the columns of $\mathbf{G} = [\mathbf{g}_1, \ldots, \mathbf{g}_c]$ are cluster prototypes which are computed as average feature vectors according to

$$\mathbf{g}_k = \sum_{j=1}^{K} (\mathbf{U}_{kj})^f \mathbf{y}_j / \sum_{j=1}^{K} (\mathbf{U}_{kj})^f, \quad k = 1, \ldots, c. \tag{15}$$

$\mathbf{V}$ is a matrix of Euclidean squared distances between prototypes and feature vectors, i.e.

$$\mathbf{V}_{kj} = \|\mathbf{g}_k - \mathbf{y}_j\|^2. \tag{16}$$

The algorithm is initialized by a random fuzzy partition $\mathbf{U}$ satisfying (13). Then it alternatively updates the prototypes via (15) and the affiliation matrix $\mathbf{U}$ via

$$\mathbf{U}_{kj} = \Big(\sum_{i=1}^{c} (\mathbf{V}_{kj}/\mathbf{V}_{ij})^{1/(f-1)}\Big)^{-1} \tag{17}$$

until convergence is achieved.

In the relational description case, the feature vectors $\mathbf{y}_1 \ldots \mathbf{y}_K$ are not available, so the cluster prototypes cannot be explicitly defined. Instead, a matrix $\mathbf{B}$ of pair-wise distances (dissimilarities) between the clustered objects is given. RFCM was derived based on the assumption that the distances in $\mathbf{B}$ are Euclidean. This allows to express $\mathbf{V}_{kj}$ as a function of $\mathbf{U}$ and $\mathbf{B}$ as

$$\mathbf{V}_{kj} = (\mathbf{B}\boldsymbol{\mu}_k)_j - \frac{1}{2}\boldsymbol{\mu}_k^T \mathbf{B}\boldsymbol{\mu}_k, \tag{18}$$

where

$$\boldsymbol{\mu}_k = [(\mathbf{U}_{k1})^f, \ldots, (\mathbf{U}_{kK})^f]^T / \sum_{j=1}^{K} (\mathbf{U}_{kj})^f. \tag{19}$$

Hence, RFCM proceeds by updating, respectively, (15), (19) and (18) until convergence is achieved.

The assumption that the distances in $\mathbf{B}$ are Euclidean need not be fulfilled when applying RFCM (as in our case). Then it might happen that (18) yields negative values. In order to avoid this problem and to guarantee the convergence of RFCM to a meaningful result, a *spreading* transform is applied to $\mathbf{V}$ and $\mathbf{B}$. It consists in adding a positive number φ to off-diagonal elements of $\mathbf{B}$ if $\mathbf{V}_{kj} < 0$ for any $k = 1 \ldots c, j = 1 \ldots mL$, that is

$$\mathbf{B}_{ij} \leftarrow \begin{cases} \mathbf{B}_{ij} + \varphi & i \neq j \\ 0 & i = j. \end{cases} \tag{20}$$

The computation of optimum φ involves a computationally expensive eigenvalue problem. On the other hand, a gross overestimate of φ may result in the distortion of $\mathbf{B}$ and a possible loss of cluster information. The paper [13] proposes a sufficiently accurate approximation of φ. It uses the by-products of the calculation of $\mathbf{V}$, which lowers the computational demands. If $\mathbf{V}_{kj} < 0$ for any $k = 1 \ldots c, j = 1 \ldots mL$,

$$\varphi = \max\{-2(\mathbf{V}_{kj})/(\|\boldsymbol{\mu}_k - \mathbf{e}_j\|^2)\}, \tag{21}$$

where $\mathbf{e}_j$ is the jth column of the $mL \times mL$ identity matrix. Subsequently, the substitution

$$\mathbf{V}_{kj} \leftarrow \mathbf{V}_{kj} + \varphi\,\|\boldsymbol{\mu}_k - \mathbf{e}_j\|^2 \tag{22}$$

is performed for $k = 1, \ldots, c$ and $j = 1, \ldots, mL$ along with (20).

What is left is to define the matrix $\mathbf{B}$ of pair-wise distances (dissimilarities) between ICs. The distances should be derived from the similarity matrix $\mathbf{D}$, which can be done in multiple ways. We define non-diagonal elements of $\mathbf{B}$ as $\mathbf{B}_{ij} = 1/\mathbf{D}_{ij}$, and the diagonal elements are put equal to zero.

B. Weighting

Once the components are clustered and the matrix of fuzzy affiliations $\mathbf{U}$ is given, the weights (the diagonal elements) in $\mathbf{\Lambda}$ can be defined accordingly. We propose the following novel schemes.

1) Indirect application: This approach transforms the fuzzy partitioning $\mathbf{U}$ into a hard one by assigning each component to the cluster with the highest affiliation degree, and the binary weights are taken as in (4). In other words, the fuzzy clustering algorithm is used to determine the hard partitioning.

2) Modified application: The weights $\mathbf{\Lambda}$ are defined by

$$\mathbf{\Lambda}_{kj} = \left(\frac{\mathbf{U}_{kj}}{1 - \mathbf{U}_{kj}} \right)^{\alpha}, \tag{23}$$

where α is a positive adjustable parameter. We use $\alpha = 2$. This approach exploits the fuzzy partitioning obtained from $\mathbf{U}$ in full.

3) Direct application: The fuzzy affiliations $\mathbf{U}_{kj}$ are used directly as the weights through

$$\mathbf{\Lambda}_{kj} = (\mathbf{U}_{kj})^{\alpha}. \tag{24}$$

where $\alpha = 2$.

V. Experiments

The experiments described here were designed to compare the clustering algorithms, weighting schemes and similarity criteria when applied within the T-ABCD algorithm. The experimental data used in all experiments consist of two different sounds that were played simultaneously over loudspeakers and recorded by eight microphones. The experiments were done an ordinary room that is described by Figure 1. Five combinations of six distinct sources including two male voices, two female voices, a typewriter sound, and a Gaussian noise were considered.

The separation quality is evaluated using the BSS_EVAL toolbox [16] in terms of three criteria: (i) Signal-to-Interference ratio (SIR), (II) Signal-to-Distortion ratio (SDR) and (iii) Signal-to-Artifact ratio (SAR).

A. Clustering of independent components

The experiment described here compares the relational fuzzy clustering techniques described in section IV and the SAHN algorithm when they are applied within T-ABCD to cluster the ICs obtained by ICA. To this end, an oracle hard clustering is derived using known SIR of ICs, which can be computed using a priori known responses of the original sources. This clustering serves as a reference one and is compared with results of the other clustering algorithms.

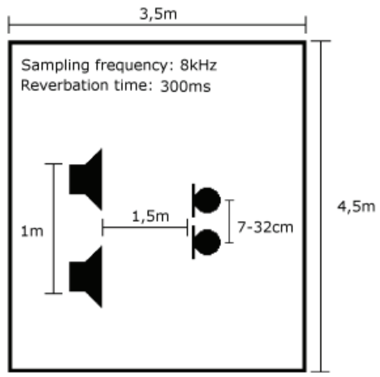

Fig. 1. Scheme of the room where the mixtures were recorded.

The reference clustering is defined as the one that assigns a component to the source (cluster) that embodies the highest SIR. Let $\mathbf{w}_i^T$ be the ith row of the de-mixing matrix $\mathbf{W}$, so $\mathbf{w}_i^T \mathbf{X}$ is the ith component. Then, the SIR of the component subject to the kth source is equal to

$$\text{SIR}_i^k = \frac{\mathbf{w}_i \mathbf{S}_k \mathbf{S}_k^T \mathbf{w}_i^T}{\mathbf{w}_i (\mathbf{X} - \mathbf{S}_k)(\mathbf{X} - \mathbf{S}_k)^T \mathbf{w}_i^T}, \tag{25}$$

where $\mathbf{S}_k$ is defined in (6).

Assuming that the reference clustering described above gives correct decisions, these decisions were compared to the ones obtained by the SAHN method. To allow the comparison with the results of the fuzzy techniques, the fuzzy partition was transformed into a hard one so that each component was assigned to the cluster/source where it achieved the highest fuzzy affiliation.

The experiment was conducted on two-microphones recordings, which are obtained when only two of the eight microphones are selected. Since there are $7 \cdot 8/2 = 28$ possibilities how to choose the microphones, we get $5 \cdot 7 \cdot 8/2 = 140$ different separation scenarios, because there are five different combinations of the signals.

Now, when separating each two-microphone mixture by T-ABCD, there are $2L$ components obtained be ICA that should be clustered. In total, there are $140 \cdot 2L$ clustering decisions.

Table I shows the number of incorrect decisions obtained by the methods for various separating filter lengths L. The fuzzyfication parameter was in this experiment set to $f = 1.5$. The experiment shows that RFCM achieves the lowest number of incorrect assignments compared to all other algorithms for all available filter lengths L.

The computational burden of the compared clustering algorithms was measured in terms of the time needed to complete all 140 clustering tasks. The experiment was performed in Matlab 7.9 on a PC with a double-core 2,66GHz processor and 2GB RAM. The resulting times are shown in Table II. As can be seen, the iterative fuzzy algorithms are almost five times faster than SAHN.

TABLE I
THE NUMBER OF INCORRECT COMPONENT ASSIGNMENTS COMPARED TO THE REFERENCE (ORACLE) CLUSTERING

	L=16	L=21	L=26	L=31
SAHN	610	782	988	1260
RFCM	599	770	958	1232
RFCMdd	763	1087	1458	2108
RPCM	1530	1971	2520	3048
Total decisions	4480	5880	7280	8680

TABLE II
THE TIME (IN SECONDS) NEEDED TO PERFORM 140 CLUSTERING TASKS

	L=16	L=21	L=26	L=31
SAHN	13.34	13.01	13.42	14.53
RFCM	2.77	2.85	2.91	2.97
RFCMdd	2.36	2.36	2.38	2.55
RPCM	3.70	3.74	3.77	3.85

B. Comparison of Weighting Schemes

The experiment described here is designed to evaluate the performance of T-ABCD when different weighting schemes are applied together with the SAHN and RFCM clustering algorithms. The parameters in T-ABCD are here set to $L = 16$, $N = 6000$, $f = 1.2$, and $\alpha = 2$. In case of the SAHN clustering algorithm, the weights are computed according to (4), while RFCM applies the weightings introduced in Section IV.B. The results in terms of SIR, SDR and SAR averaged over all scenarios and separated signals are shown in Table III.

The experiment suggests that the separation results of T-ABCD with hierarchical clustering and T-ABCD with indirectly determined weights are nearly identical. This means that both clustering techniques are similarly successful when clustering the ICs, because they determine the weights in the same manner. Slightly better results are obtained by the direct application of the affiliations $\mathbf{U}$ to reconstruction weights.

The modified computation of the reconstruction weights outputs estimates which are characterized by high values of SIR by low values of SAR. A subjective listening to the estimated signals confirms that the signals are fairly well separated but heavily distorted.

It can be concluded that the utilization of RFCM within T-ABCD and the direct application of affiliations $\mathbf{U}$ as reconstruction weights result in a significant improvement of the performance. Moreover, the RFCM algorithm is less computationally demanding than SAHN (provided that the number of clusters/sources is known) and has a favorable iterative computation scheme. This feature is advantageous when time-varying mixing scenario is considered, where continuous updates of the separating filters are needed; see [18]. In such case, the clustering algorithm can be initialized by its previous solution, and one iteration is performed each time the affiliations of ICs are updated.

TABLE III
AVERAGE SEPARATION PERFORMANCE OF T-ABCD WITH DIFFERENT CLUSTERING AND WEIGHTING TECHNIQUES

	SIR[dB]	SDR[dB]	SAR[dB]
SAHN	7.56	5.74	12.69
RFCM - Indirect	7.53	5.72	12.71
RFCM - Direct	7.73	5.98	13.12
RFCM - Modified	11.92	5.35	7.11

TABLE IV
PERFORMANCE OF T-ABCD USING DIFFERENT SIMILARITY CRITERIA

	SIR[dB]	SDR[dB]	SAR[dB]
SAHN (Proj.)	10.64	6.16	9.28
RFCM (Proj.)	7.93	5.92	12.68
SAHN (GCC)	7.56	5.74	12.69
RFCM (GCC)	7.73	5.98	13.12
SAHN (Coh.)	7.66	5.85	12.94
RFCM (Coh.)	8.02	6.29	13.82

C. Comparison of Similarity Criteria

The goal here is to compare separation results of T-ABCD when the criteria of similarity described by Section III are used. The experiment is done in the same way as the previous one. The results averaged over all scenarios and separated signals are shown in Table IV.

The results indicate that all considered measures are suitable as basis for clustering of ICs in T-ABCD. The best separation performance was achieved using the coherence (12). The projection approach along with utilization of SAHN hard clustering leads to well separated mixtures at the cost of stronger presence of artifacts in the estimated sources.

For all similarity measures, the usage of fuzzy clustering via the RFCM method leads to higher separation performance.

VI. CONCLUSIONS

The utilization of the fuzzy algorithm RFCM for the clustering of ICs within blind audio source separation method T-ABCD was proposed and described in details. Three measures for similarity of ICs were introduced, specifically, the projections, generalized correlations and coherences. All clustering techniques and the similarity measures were compared in experiments.

The T-ABCD variant endowed by the fuzzy grouping yields an improved separation performance and is less computationally demanding compared to the original T-ABCD using SAHN. The advantageous iterative scheme of RFCM allows its utilization in T-ABCD variant designed for time-varying mixtures.

All three proposed similarity measures are suitable for utilization in T-ABCD. In the experimental comparison, the coherence approach along with fuzzy clustering is shown to be the most suitable choice.

ACKNOWLEDGMENT

This work was supported by Grant Agency of the Czech Republic through the project P103/11/1947 and by the Student

Grant Scheme (SGS) at the Technical University of Liberec.

References

[1] Z. Koldovský and P. Tichavský, Time-Domain Blind Separation of Audio Sources on the basis of a Complete ICA Decomposition of an Observation Space, *IEEE Trans. on Speech, Audio and Language Processing*, Vol. 19, No. 2, pp. 406-416, Feb. 2011.

[2] J. Málek, Z. Koldovský, J. Žďánský, J. Nouza, Enhancement of Noisy Speech Recordings using Blind Source Separation, *Proc. Interspeech 2008*, Brisbane, Austrlie, 2008.

[3] Z. Koldovský, P. Tichavský, "Time-domain blind audio source separation using advanced component clustering and reconstruction," *Proc. of HSCMA 2008*, Trento, Italy, pp. 216-219, May 2008.

[4] P. Comon, Ch. Jutten, *Handbook of Blind Source Separation: Independent Component Analysis and Applications*, Elsevier, Academic Press, 2010.

[5] A. Hyvärinen, J. Karhunen, E. Oja, *Independent component analysis*, Wiley-Interscience, New York, 2001.

[6] S. C. Douglas, H. Sawada, and S. Makino, "Natural gradient multichannel blind deconvolution and speech separation using causal FIR filters," *IEEE Trans. Speech Audio Processing*, Vol. 13, No. 1, pp. 92-104, Jan. 2005.

[7] H. Buchner, R. Aichner, and W. Kellermann, "A Generalization of Blind Source Separation Algorithms for Convolutive Mixtures Based on Second-Order Statistics, " *IEEE Trans. on Speech and Audio Processing*, Vol. 13, No. 1, pp. 120-134, Jan. 2005.

[8] Z. Koldovský and P. Tichavský, "Time-Domain Blind Audio Source Separation Using Advanced ICA Methods, " *Proc. Interspeech 2007*, pp. 846-849, August 2007.

[9] Ch. H. Knapp, G. C. Carter, "The Generalized Correlation Method for Estimation of Time Delay," *IEEE Transactions on Signal Processing*, Vol. 24, No. 4, 1976.

[10] T. Hastie, R. Tibshirani, J. Friedman, *The Elements of Statistical Learning (2nd ed.)*, New York: Springer. pp. 520528, 2009.

[11] J. V. Oliveira , W. Pedrycz, *Advances in Fuzzy Clustering and its Applications*, Wiley-Interscience, 2007.

[12] R. Krishnapuram, A. Joshi, O. Nasraoui, L. Yi, "Low-Complexity Fuzzy Relational Clustering Algorithms for Web Mining, " *IEEE Transactions on Fuzzy Systems*, Vol. 9, No. 4, pp. 595 - 607, 2001.

[13] R. J. Hathaway, J. C. Bezdek, J. W. Davenport, "On relational data versions of c-means algorithm, " *Elsevier - Pattern recognition letters*, No. 17, pp. 607-612, 1996.

[14] J. C. Bezdek, Fuzzy Mathematics in Pattern Classification, *PhD Thesis at Cornell University*, Ithaca, USA, 1973.

[15] P. Tichavský and A. Yeredor, "Fast Approximate Joint Diagonalization Incorporating Weight Matrices, " *IEEE Transactions of Signal Processing*, Vol. 57, No. 3, pp. 878-891, 2009.

[16] C. Févotte, R. Gribonval, and E. Vincent, "BSS EVAL toolbox user guide, " IRISA, Rennes, France, Tech. Rep. 1706, 2005, [Online], `http://www.irisa.fr/metiss/bss eval/`

[17] C. Févotte, A. Debiolles, and C. Doncarli. "Blind separation of FIR convolutive mixtures: application to speech signals," *Proc. 1st ISCA Workshop on Non-Linear Speech Processing*, 2003.

[18] J. Málek, Z. Koldovský and P. Tichavský, "Adaptive Time-Domain Blind Separation of Speech Signals," in Latent Variable Analysis and Signal Separation, *Lecture Notes in Computer Science*, Volume 6365, pp. 9-16, Springer, 2010.