

Technical University of Liberec  
Faculty of Mechatronics, Informatics and  
Interdisciplinary Studies  
Institute of Information Technology and Electronics



Blind audio source separation  
via  
Independent Component Analysis

*Doctoral Thesis Review*

December 2010

Ing. Jiří Málek



# Blind Audio Source Separation via Independent Component Analysis

---

Doctoral Thesis Review

Ing. Jiří Málek

Ph.D. Program: Technical Cybernetics

Branch of Study: Electronics and Informatics

Workplace: Institute of Information Technology and Electronics  
Faculty of Mechatronics, Informatics and Interdisciplinary  
Studies  
Technical University of Liberec

Supervisor: Ing. Zbyněk Koldovský Ph.D.

# Abstract

The reviewed thesis deals with the general problem of Blind Source Separation (BSS), the main focus being on audio sources. In an appendix, a research outside the topic of BSS is discussed: The automatic classification of biomedical signals.

The general goal of the blind source separation is to estimate unknown source signals from their measured mixtures, while there is no information about the mixing process available. This task is too general and cannot be solved without additional assumptions about the unknown sources. One such assumption is the mutual independence of the source signals. Methods performing the separation based on the independence assumption are referred to as Independent Component Analysis (ICA).

The thesis consists of three main parts. The first part deals with the separation of non-stationary instantaneous mixtures. The author contributed to the proposal of an ICA algorithm called Block EFICA. This algorithm is proved to be asymptotically efficient (i.e. gives the most accurate estimates of the sources possible) provided that the source signals are block-wise stationary and have constant variances.

The second and third part concerns the separation of convolutive mixtures of audio sources. This problem can be solved via ICA in the time domain, for example by an existing method known as T-ABCD, which was proposed Koldovský and Tichavský. Modifications for this method are presented. These modifications improve the reconstruction of the sources in the case of imperfect ICA separation. Furthermore, an "online" version of the T-ABCD algorithm is proposed. It is able to adapt to the changes in the mixing system and to the non-stationarity of the source signals.

The appendix of the thesis deals with automatic classification of medical signals which originate in the screening examination of lower limbs arteries. This examination, performed via inexpensive ultrasound units, aims at an early diagnostics of Peripheral Arterial Disease. Automatic classifiers for the data measured by mentioned ultrasound units are designed and trained. These classifiers are able to assign the signals to predefined classes, which reflect the degree of arterial occlusion.

# Contents

<b>1</b>	<b>Introduction</b>	<b>2</b>
1.1	Blind source separation . . . . .	2
1.2	Motivation and goals of the thesis . . . . .	4
<b>2</b>	<b>Linear Independent Component Analysis</b>	<b>4</b>
2.1	The goal of ICA . . . . .	4
2.2	Design of an ICA algorithm . . . . .	5
2.3	Existing ICA methods and their statistical principles . . . . .	5
2.4	The EFICA algorithm . . . . .	6
<b>3</b>	<b>The Block EFICA algorithm</b>	<b>6</b>
3.1	Parametric score function estimator . . . . .	7
3.2	Separation of block-wise stationary sources . . . . .	7
3.3	Experiments with the Block EFICA algorithm . . . . .	8
<b>4</b>	<b>Separation of Convolutive Mixtures</b>	<b>10</b>
4.1	Separation via ICA in the time-domain . . . . .	10
4.2	The T-ABCD algorithm . . . . .	12
<b>5</b>	<b>Author's Modifications of the T-ABCD Algorithm</b>	<b>13</b>
5.1	RFCM applied in T-ABCD . . . . .	14
5.2	Experiments . . . . .	14
<b>6</b>	<b>The Online T-ABCD algorithm</b>	<b>15</b>
6.1	Description of the algorithm . . . . .	16
6.2	Experiments . . . . .	19
<b>7</b>	<b>Conclusions and Future Work</b>	<b>20</b>
7.1	Separation of non-stationary non-gaussian sources . . . . .	21
7.2	Modifications of the T-ABCD algorithm . . . . .	21
7.3	Adaptive separation of audio signals . . . . .	21
7.4	Future work . . . . .	22
	<b>Appendix: Automatic Classifiers for Medical Data from Doppler Unit</b>	<b>23</b>
8.1	Introduction and motivation . . . . .	23
8.2	Automatic Classifier Design . . . . .	24
8.3	Experiments . . . . .	27
8.4	Conclusions . . . . .	27
	<b>Bibliography</b>	<b>28</b>

# 1. Introduction

## 1.1 Blind source separation

The general goal of Blind Source Separation (BSS, [1]) is to estimate unknown sources from a set of observed mixtures. The estimation is performed with no prior information about either the sources or the mixing process. Specific restrictions are placed on the mixing model and the source signals in order to limit the generality.

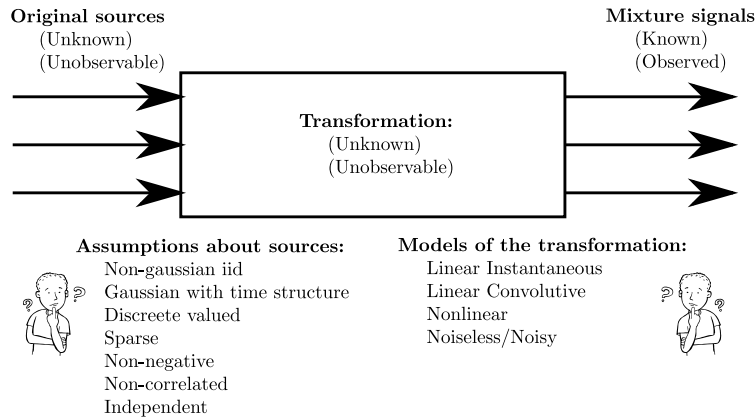


Figure 1.1: The general blind separation problem

### Models of the mixing process

The **linear instantaneous model** is encountered in situations where the sampling frequency is small in comparison to the propagation speed of the signals. Here, all the signals reach the sensors at the same time.

The linear instantaneous model defines observed mixtures as a weighted sum of unknown source signals. Such transformation is then described as

$$\mathbf{X} = \mathbf{A} \cdot \mathbf{S}, \quad (1.1)$$

where  $\mathbf{X}$  is the known  $m \times N$  matrix of mixtures,  $m$  is the number of mixtures obtained by measurement on sensors and  $N$  is the number of available samples. Matrix  $\mathbf{S}$  contains the samples of the original unknown sources and is of size  $d \times N$ ,  $d$  is the number of unknown sources. A single row of  $\mathbf{X}$  or  $\mathbf{S}$  will be further denoted as  $\mathbf{x}_i$  or  $\mathbf{s}_i$ , a single column as  $\mathbf{X}(n)$  or  $\mathbf{S}(n)$  and a single element as  $x_i(n)$  or  $s_i(n)$ , respectively. The unknown  $m \times d$  matrix  $\mathbf{A}$  represents parameters of the mixing model.

The **linear convolutive mixing model** is a generalization of (1.1), because the observed mixtures contain *delayed* original sources and their echoes. This problem is often encountered in acoustics, where it is called the *cocktail party problem*. It

can be described by

$$x_i(n) = \sum_{k=1}^d \sum_{\tau=0}^{M_{ik}} a_{ik}(\tau) \cdot s_k(n - \tau), \quad (1.2)$$

where  $x_1(n) \dots x_m(n)$  are the signals observed on microphones,  $s_1(n) \dots s_m(n)$  are the original unknown sources. The unknown parameters  $a_{ik}(\tau)$  represent the source-sensor impulse responses, i.e. impulse responses expressing the propagation of sound from the location of each source to each microphone. The convolutive model coincides with the instantaneous model, when  $M_{ik} = 0$ .

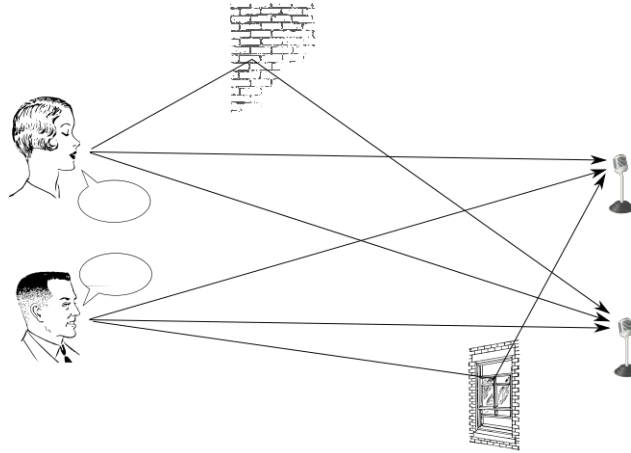


Figure 1.2: An example of speech signal propagation in the real environment.

## Source models and BSS methods

In order to allow the separation, general assumptions are placed on the unknown sources. These assumptions determine techniques which can be exploited for source estimation. The reviewed thesis utilizes following separation methods:

**Independent Component Analysis** [2] assumes that the unknown sources are statistically independent random processes. The ICA techniques model the sources according to one of the two following principles.

The sources can be considered as *non-gaussian and temporally independent and identically distributed* random variables (iid). This leads to the utilization of higher (mostly 4<sup>th</sup>) order statistics.

The sources can be modeled as *gaussian with certain temporal structure* (temporally dependent). This leads to the utilization of second order statistics.

**Principal Component Analysis** [3] assumes that the signals  $\mathbf{s}_i$  are uncorrelated and have the highest variance among all orthogonal transforms of the mixtures  $\mathbf{X}$ . Since, the uncorrelatedness is a necessary condition of independence, the PCA is often used as preprocessing step prior to the utilization of ICA.

## 1.2 Motivation and goals of the thesis

Firstly, the thesis deals with the **separation of instantaneous mixtures** via ICA. Although this problem has been studied since the early 1990s and is now well explored, there are still areas which deserve more research attention. Most of the ICA methods assume that the sources are either iid random variables or Gaussian random processes with a certain temporal structure. Although these models cover many types of signals, there exist very important sources which do not follow these assumptions. For example, speech is often modeled by Laplace distribution. Moreover, speech is stationary only in short intervals of 20-25 ms. This fact motivates the proposal of an ICA method, which is able to efficiently separate the non-stationary non-Gaussian sources. The algorithm design is discussed in Chapter 3.

Subsequently, the thesis deals with the **blind separation of real-world audio sources**. Nowadays, this task is widely studied and has considerable practical utilization. The main challenge here poses the separation in highly reverberant environments or the separation in presence of the background noise. The mixing of audio sources is usually modeled by the convolutive mixture model. Thanks to the plausible assumption of the independence of the audio sources, the problem can be solved via ICA. However, the convolutive mixture needs to be transformed into the instantaneous one first.

Recently, a method denoted as T-ABCD [4] has been proposed by Koldovský and Tichavský. It solves the separation task via ICA in the time-domain. The method exhibits an advantageous modular structure. This allows simple implementation of modifications which improve the performance of the method. The original T-ABCD assumes a perfect separation performed by ICA and reconstructs the sources accordingly. This is usually not true in practice. Chapter 5 discusses the modification, which aims at improvement of the reconstruction of the sources in the case of imperfect ICA separation.

The original T-ABCD assumes that the mixing system is fixed and does not change its inner parameters in time. This fact may not be entirely true in practice, e.g. due to the changing positions of the sources. This motivates the proposal of a new version of T-ABCD, further denoted as online, which is able to adapt to the changes in parameters of the mixing system. This subject is presented in Chapter 6.

## 2. Linear Independent Component Analysis

### 2.1 The goal of ICA

Let us focus on the case when  $d = m$  in (1.1), which is referred to as the **determined mixtures**. Here, the matrix  $\mathbf{A}$  is square and regular. The goal of ICA is to estimate the demixing matrix  $\mathbf{W} = \mathbf{A}^{-1}$  so that the estimated signals

$$\hat{\mathbf{S}} = \widehat{\mathbf{W}} \cdot \mathbf{X}, \quad (2.1)$$

are as independent as possible. The matrices  $\hat{\mathbf{S}}$  and  $\widehat{\mathbf{W}}$  denote the estimates of  $\mathbf{S}$  and  $\mathbf{W}$ , respectively. The rows of  $\hat{\mathbf{S}}$  are called *Independent Components* (ICs).



## 2.2 Design of an ICA algorithm

In most cases, the ICA methods can be divided into three distinct parts. Prior to the application of ICA on observed mixtures  $\mathbf{X}$ , some *preprocessing* technique may be exploited. This includes e.g. the removal of the sample mean or decorrelation of the mixtures. Let us denote the preprocessed mixtures by  $\mathbf{Z}$ .

The ICA algorithm itself often consists of the formulation of a certain *objective (contrast) function*, which is subsequently optimized via an *optimization technique*. The contrast function is used to quantify the mutual dependence of components and its global extrema matches the solution of the separation problem.

The ICA methods differ in the manner the components are estimated. The mixtures can be available upon the execution of the method, such algorithms are called *Batch*. The other possibility is that the algorithm acquires the data in successive steps during the computation and the separating matrix  $\widehat{\mathbf{W}}$  is continuously updated. These methods are denoted as *Online*.

## 2.3 Existing ICA methods and their statistical principles

Many ICA algorithms assume the unknown sources in a form of **iid random variables**. Here, the separation is carried out through the optimization of some criterion based on higher order statistics. These methods assume that not more than one of the sources has Gaussian distribution. It stems from the fact that any orthogonal transform applied on two independent Gaussian distributions results in two independent Gaussian variables.

Following well known methods are based on the iid source model. An algorithm known as Infomax, based on maximum likelihood approach and using parametric density approximation, was proposed by Bell and Sejnowski in [5]. A fixed-point algorithm called FastICA, based on approximation of entropy, was introduced by Hyvärinen and Oja in [6]. The properties of cumulants and cumulant based tensors were utilized for the contrast function formulation by Cardoso, who proposed the algorithm JADE (Joint Approximate Diagonalization of Eigenmatrices) [7].

However, many real-world sources are not iid and exhibit some form of **temporal structure**. Methods for separation of these sources exploit second order statistics and are able to separate Gaussian signals. Following two types of temporal structure are considered most often.

Spectral diversity based methods assume sources to be *Gaussian wide-sense stationary processes* (WSS). An algorithm called SOBI (Second Order Blind Identification) based on the diagonalization of several time-lagged covariance matrices, was introduced by Belouchrani et al. in [8].

The non-stationarity based methods assume the unknown sources in the form of *Gaussian random variable with a changing variance*. An algorithm called BGL (Block Gaussian Likelihood), designed for the separation of block-wise stationary Gaussian sources, was proposed by Pham and Cardoso in [9].

## 2.4 The EFICA algorithm

The EFICA algorithm is an extension of well known ICA algorithm FastICA [6]. It is designed for separation of instantaneous mixtures of iid sources and is proven to be asymptotically efficient under the assumption that the probability distribution of the independent components belongs to the class of *Generalized Gaussian Distribution* (GGD) with the parameter  $\alpha < 2$ .

**EFICA** measures the independence by the entropy approximations in the form of suitable non-linear functions  $G(\cdot)$ .

If the densities  $p_i(x)$  were known, the function  $G_i^{opt}(x) = -\log(p_i(x))$  would be optimal, because  $-\mathbb{E}\{\log(p_i(x))\}$  is the entropy itself. Therefore, the log-densities of some known important probability distributions are often utilized as  $G(x)$ . Derivative of  $G_i^{opt}(x)$  given by  $g_i^{opt}(x) = [G_i^{opt}(x)]' = -p_i'(x)/p_i(x) = \psi_i(x)$  is referred to as *the score function* of  $p_i(x)$ . The source distributions are unknown, but EFICA models all sources as if they have GGD with an appropriate parameter  $\alpha$ .

The algorithm consists of three steps. In the first step, consistent pre-estimates of the sources via **(I) the symmetric FastICA** with standard nonlinearity  $g(x) = \tanh(x)$ .

The second step resides in **(II) the adaptive choice of nonlinearity**  $G_i(x)$ . It is assumed that the sources are distributed according to GGD. The free parameter  $\alpha$  is estimated by equaling the sample fourth-order moment of the pre-estimated sources by the theoretical fourth-order moment of the GGD. Subsequently, an optimal function  $G_i^{opt}(x)$  is selected for the estimated source distribution  $\text{GGD}(\alpha)$ .

The final step of the algorithm is called **(III) the refinement**. Here, several one-unit FastICA iterations given by

$$\begin{aligned} \mathbf{w} &\leftarrow \hat{\mathbb{E}}\{\mathbf{Z}g(\mathbf{w}^T\mathbf{Z})\} - \hat{\mathbb{E}}\{g'(\mathbf{w}^T\mathbf{Z})\mathbf{w}\} \\ \mathbf{w} &\leftarrow \mathbf{w} / \|\mathbf{w}\|, \end{aligned} \tag{2.2}$$

are performed for each of the source estimates from step (I) with the nonlinearity selected in step (II). These iterations are called *fine-tuning*. Here,  $\mathbf{w}$  denotes rows of the demixing matrix  $\mathbf{W}$  and  $\hat{\mathbb{E}}$  is the sample mean operator.

Finally, auxiliary constants  $c_{k\ell}; k, \ell = 1 \cdots d$  are computed which minimize the mean square error of the rows of  $\widehat{\mathbf{W}}$  so that the algorithm attains the asymptotical efficiency. Details on EFICA can be found in [10] or within the reviewed thesis.

## 3. The Block EFICA algorithm

The Block EFICA algorithm is designed for the separation of non-stationary and non-gaussian sources. The algorithm is proven to be asymptotically efficient (i.e. gives asymptotically the most accurate estimates of the sources possible), provided that the variances of unknown sources are constant. The algorithm is based on the EFICA algorithm [10]. There are two major differences between EFICA and Block EFICA algorithms.

The first modification consists in the implementation of a general parametric score function estimator [a3]. The estimated score functions are used as contrasts in the one-unit iterations within the fine-tuning part of the method.

The second realized modification consists in the introduction of non-stationarity into the EFICA's model [a4, a6]. The Block EFICA assumes the sources to be block-wise stationary.

### 3.1 Parametric score function estimator

The original EFICA implements the score function estimator for the GGD sources. This section considers more general score function estimator which is not restricted for GGD sources and thus can be more accurate for a wider range of distributions.

The proposed modification is following. Within the adaptive second step of EFICA, the function  $g_k(x)$  is selected as a score function estimate for each of the sources separately. This is done prior to the first one-unit iteration and the non-linearity is re-estimated after each iteration.

The utilized estimator [12] minimizes the criterion given as the mean squared error between the true unknown score function  $\Psi(y)$  and its searched estimate  $\mathbf{h}(y|\theta)$ , i.e.

$$E[(\Psi(y) - \mathbf{h}(y|\theta))^2], \quad (3.1)$$

where  $\theta$  is the vector of parameters which is subject to minimization and  $E$  stands for the expected value operator.

To simplify the optimization problem, the function  $\mathbf{h}(y|\theta)$  is considered to be a linear combination of suitable basis functions  $h_1(y), \dots, h_k(y)$ , i.e.  $\mathbf{h}(y|\theta) = \sum_{i=1}^k \theta_i \cdot h_i(y)$ . A suitable choice of basis functions is discussed in the reviewed thesis.

### 3.2 Separation of block-wise stationary sources

In the case of non-stationary source signals, the samples of  $\mathbf{s}_i$  are not identically distributed. To allow practical estimation, it is expected that there are  $Q$  blocks in  $\mathbf{S}$  of the same integer length. In these blocks, the distribution of the signals does not change. In this section, the upper index ( $I$ ) is used to denote random variables or functions which are related to the  $I$ th such block.

The source signals  $\mathbf{s}_i$  are in blocks i.i.d. random variables. Thus, the instantaneous model in (1.1) holds in each considered block, i.e.

$$\mathbf{X}^{(I)} = \mathbf{A} \cdot \mathbf{S}^{(I)}, I = 1 \dots Q. \quad (3.2)$$

The Block EFICA consist of three steps, which are similar to those in the EFICA algorithm.

**(I) The symmetric FastICA** [6] is exploited to obtain the pre-estimate of the demixing matrix  $\widehat{\mathbf{W}}$ .

(II) **The fine-tuning** of the rows of  $\widehat{\mathbf{W}}$  is performed. The computation proceeds for each of the sources separately via modified FastICA one-unit iterations given by

$$\begin{aligned} \mathbf{w}_k \leftarrow & \lambda_k^{(1)} (\mathbb{E}[\mathbf{Z}^{(1)} g_k^{(1)}(\mathbf{w}_k^T \mathbf{Z}^{(1)})] - \mathbf{w}_k \mathbb{E}[g_k^{(1)' }(\mathbf{w}_k^T \mathbf{Z}^{(1)})]) + \dots \\ & \dots + \lambda_k^{(Q)} (\mathbb{E}[\mathbf{Z}^{(Q)} g_k^{(Q)}(\mathbf{w}_k^T \mathbf{Z}^{(Q)})] - \mathbf{w}_k \mathbb{E}[g_k^{(Q)' }(\mathbf{w}_k^T \mathbf{Z}^{(Q)})]). \end{aligned} \quad (3.3)$$

In practice, when the number of samples is finite, the expectations are replaced by sample means. Throughout the text, equation (3.3) is further referred to as *block one-unit FastICA iteration*. The non-linearities  $g_k^{(I)}(\cdot)$  are computed using the parametric score function estimator described in Section 3.1 and are re-estimated after each one-unit iteration. Simultaneously, the weights  $\lambda_k^{(I)}$  are updated, in order to optimize the performance of Block EFICA. The selection of weights  $\lambda_k^{(I)}$  is based on the theoretical performance analysis, which is discussed in details in the reviewed thesis.

(III) **The refinement** consists in the computation of the auxiliary constants  $c_{k\ell}, k, \ell = 1 \dots d$  which minimize the mean square error of rows of  $\widehat{\mathbf{W}}$ . In this manner, the asymptotical efficiency of the algorithm is attained. The constants are exploited to form the matrix

$$\mathbf{W}_k^+ = [c_{k1} \mathbf{w}_1 / \|\mathbf{w}_1\|, \dots, c_{kd} \mathbf{w}_d / \|\mathbf{w}_d\|]^T \quad (3.4)$$

Then, the  $k$ -th row of the matrix  $(\mathbf{W}_k^+ \mathbf{W}_k^{+T})^{-1/2} \mathbf{W}_k^+$  is the  $k$ th row of the final demixing matrix  $\widehat{\mathbf{W}}^{\text{BEF}}$  estimated by Block EFICA. The selection of constants  $c_{k\ell}$  is as well based on the theoretical performance analysis and is discussed in detail in the reviewed thesis.

### 3.3 Experiments with the Block EFICA algorithm

The first presented example deals with separation of artificial sources with constant variance which correspond to the theoretical model of Block EFICA. It verifies the accuracy and stability of the proposed algorithm.

The other experiment confirms the robustness of Block EFICA in scenarios where the respective model does not hold. This concerns the experiments with real world speech data. Moreover, the examples with real world data prove the practical contribution of the algorithm.

More experiments concerning the BlockEFICA algorithm can be found in the reviewed thesis.

#### Separation of block-wise stationary constant variance signals

The first example considers the separation of twenty artificial block-wise stationary sources. The length of the signals is  $N = 10000$  samples. Each signal consists of four blocks of the same length  $N/4$ . The first and the third blocks have the Gaussian distribution and the second and the fourth blocks have the GGD( $\alpha$ ) distribution.

The parameter  $\alpha$  is fixed for each of the 20 signals, where its values are uniformly chosen from  $[0.1, 10]$ . The variance of all the distributions is one. Thus, the signals have constant variance. For the purposes of the experiment, these signals are mixed by a random matrix  $\mathbf{A}$ . The results shown in Figure 3.1 are presented in the terms of the mean interference-to-signal ratio and are averaged over 100 Monte Carlo trials.

Results of this experiment corroborate the validity of the theoretical performance analysis stated in the reviewed thesis and [a6] due to the proximity of the theoretical results with the empirical ones. They also demonstrate the improved performance of the proposed method compared to EFICA, as different distributions on the four blocks of signals were considered.

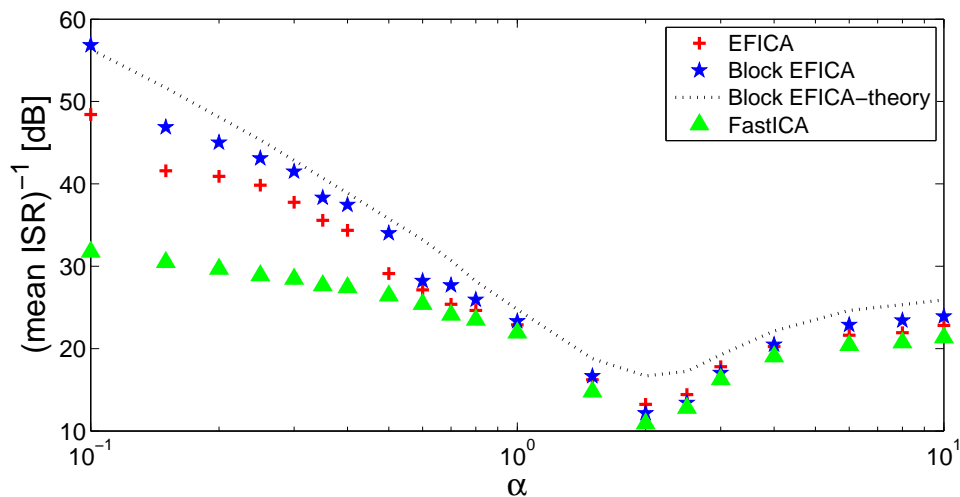


Figure 3.1: Mean interference-to-signal ratio of the separated signals in the experiment with 20 block-wise stationary sources, computed over 100 Monte Carlo trials.

### Separation of real-world speech signals mixed linearly

The proposed example demonstrates a practical contribution of the method, which is shown to be robust when separating real world sources. These sources do not follow the source model of Block EFICA (3.2).

Twenty speech signals of length 5000 samples are mixed linearly with a random mixing matrix  $\mathbf{A}$  and subsequently separated by the competing non-gaussianity-based algorithms. The speech signals were selected randomly from a database of utterances, which is along with the source code of the method available online [13].

The above described experiment was performed 1000 times and the results were averaged over all trials and signals. The results are depicted in Figure 3.2 and are quantified in the means of three distinct criteria. The proposed Block EFICA shows improved separation results compared to the original EFICA, specifically 1dB ISR improvement and 2,5dB SIR improvement.

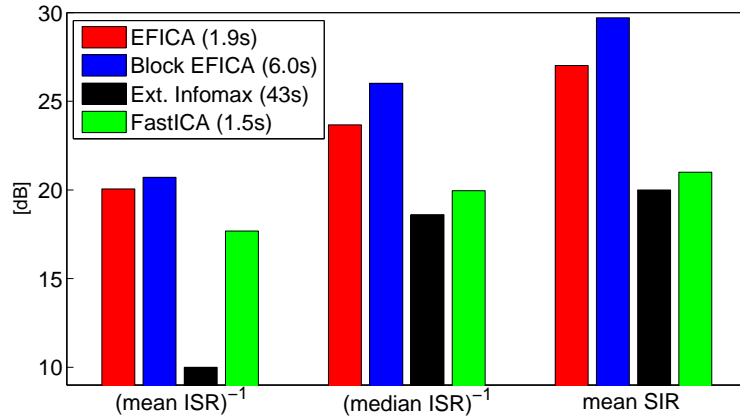


Figure 3.2: Separation of real-world speech signals mixed linearly

## 4. Separation of Convulsive Mixtures

The linear convolutive mixture model (1.2) is well suited for audio applications because it takes the propagation of sound in the real environment into consideration. Sound does not reach all microphones at the same time because its propagation speed is limited. This introduces delays into the mixing. Moreover, sound reflects on walls and other obstacles, which introduces multiple echoes of a single source into the mixing process.

The separation usually proceeds via filtering of the mixtures with some estimated MIMO filter via

$$\hat{s}_k(n) = \sum_{i=1}^m \sum_{\tau=0}^{L-1} w_{ki}(\tau) \cdot x_i(n - \tau), \quad (4.1)$$

where  $L$  is the length of the demixing MIMO filters  $w_{ki}$ .

Due to unknown filtering caused by the propagation in the environment, the sources are estimated as spatial images (responses) of the sources on microphones, rather than the sources themselves.

The response of the  $k$ th source on the  $i$ th microphone is given by

$$s_k^i(n) = \sum_{\tau=0}^{M_{ik}} a_{ik}(\tau) \cdot s_k(n - \tau). \quad (4.2)$$

To allow practical separation, a general prerequisite of the mutual independence of unknown sources is often assumed. This allows the utilization of an ICA method as a separator. Prior to the utilization of ICA, the convolutive mixing model needs to be transformed into an instantaneous one.

### 4.1 Separation via ICA in the time-domain

The transformation in the time-domain expresses discrete convolution via matrix multiplication. Two matrices  $\tilde{\mathbf{S}}$  and  $\tilde{\mathbf{X}}$ , which have (usually, not necessarily)

the block-Toeplitz structure, are introduced. The rows of these matrices represent delayed source signals or mixtures, respectively. The matrices have the size  $mL \times (N - L + 1)$ , where  $L$  is the desired length of the demixing filter. The linear space spanned by the rows of  $\tilde{\mathbf{S}}$  and  $\tilde{\mathbf{X}}$  is called source space or observation space, respectively. The matrix  $\tilde{\mathbf{X}}$  has following structure:

$$\tilde{\mathbf{X}} = \begin{bmatrix} x_1(L) & \dots & \dots & x_1(N) \\ x_1(L-1) & \dots & \dots & x_1(N-1) \\ \vdots & \vdots & \vdots & \vdots \\ x_1(1) & \dots & \dots & x_1(N-L+1) \\ x_2(L) & \dots & \dots & x_2(N) \\ x_2(L-1) & \dots & \dots & x_2(N-1) \\ \vdots & \vdots & \vdots & \vdots \\ x_m(1) & \dots & \dots & x_m(N-L+1) \end{bmatrix}, \quad (4.3)$$

which is similar for  $\tilde{\mathbf{S}}$  as well. Using this notation, the convolutive mixing model in (1.2) may be approximated via

$$\tilde{\mathbf{X}} = \tilde{\mathbf{A}}\tilde{\mathbf{S}}, \quad (4.4)$$

where  $\tilde{\mathbf{A}}$  is a matrix constructed from the elements of the source-sensor impulse responses  $a_{ik}(n)$  in (1.2).

With this approximation, it is possible to estimate the demixing filter coefficients contained in matrix  $\widehat{\mathbf{W}}$  via an ICA method described in Chapter 2.

The methods performing the separation via ICA in the time-domain can be classified based on the fact whether they perform *complete* or *partial* decomposition of the observation space  $\tilde{\mathbf{X}}$ .

**The partial decomposition** consists in the estimation of one dimensional subspaces (components). The partial decomposition approach makes the problem of rapid dimension growth of  $\tilde{\mathbf{X}}$  easier. On the other hand, these methods may find two components related with a single source and skip another source completely.

A natural gradient based method performing partial decomposition was proposed by Amari et al. in [15].

**The complete decomposition with constraints** lowers the computational demands of the methods by introducing some simplifying assumptions about the decomposing transform or the observation space.

A generalization of the ICA algorithm SOBI for convolutive mixtures, performing complete constrained decomposition, was proposed by Belouchrani et al. in [16].

**The complete unconstrained decomposition** does not introduce any simplifying assumptions. Therefore, it provides a way to utilize the available data as effectively as possible.

A modular algorithm, performing the complete unconstrained decomposition, was proposed by Koldovský and Tichavský in [4] and subsequent extensions. The algorithm is called T-ABCD, which stands for Time-domain Audio source Blind separation based on the Complete Decomposition. The algorithm is described in

Section 4.2. The modifications proposed by the author of the reviewed thesis are presented in Chapter 5.

## 4.2 Time-domain Audio Source Blind Separation Based on the Complete Decomposition

The algorithm consists of five consecutive steps. In the first step, the matrix  $\tilde{\mathbf{X}}$  in (4.3) is constructed from the known mixtures  $x_i(n)$ . The second step consists in the application of an arbitrary ICA method on  $\tilde{\mathbf{X}}$  and its decomposition into independent components (ICs). The goal of the third step is to compute a measure of similarity between ICs. This measure forms the basis for the clustering of ICs into groups corresponding to original sources. The clustering is performed in the fourth step of the algorithm. Finally, the responses of the sources on microphones are reconstructed, based on the determined clusters.

The **decomposition of the observation subspace  $\mathbf{X}$**  can be performed via an arbitrary ICA algorithm. Two algorithms were implemented as the ICA separators within T-ABCD, the EFICA algorithm (Section 2.4) and the BGSEP algorithm [11].

The result of ICA is the estimate of a demixing matrix  $\tilde{\mathbf{W}}$  denoted as  $\widehat{\tilde{\mathbf{W}}}$  and the independent components given by  $\mathbf{C} = \widehat{\tilde{\mathbf{W}}}\tilde{\mathbf{X}}$ . The  $i$ th row of  $\mathbf{C}$  is denoted as  $\mathbf{c}_i$  and its  $n$ th element as  $c_i(n)$ .

The rows of  $\tilde{\mathbf{S}}$  are not independent, due to the temporal structure of the audio/speech sources. Therefore, the components become arbitrary filtered copies of the unknown sources, i.e. the sources estimated from components exhibit random coloration.

There exist groups of ICs where all the components are related to a single original source. These groups are found via clustering, which is based on the mutual similarity of components, and are exploited for the final reconstruction of the sources.

The **similarity measure** used within the T-ABCD algorithm is based on the computation of generalized-correlation coefficients known as GCC-PHAT [18]. For the details on computation see the reviewed thesis. The mutual similarities of ICs are stored in the  $mL \times mL$  matrix  $\mathbf{D}$ .

The **clustering of independent components** is within the the original prototype of T-ABCD from [4] performed via Standard Agglomerative Hierarchical Non-overlapping algorithm (SAHN) with average linking strategy [19].

The SAHN algorithm produces so-called *Hard Partition*; it consists in computation of the affiliation matrix  $\mathbf{U}$  of size  $d \times mL$ , which describes the affiliation of each component to each cluster. Each element of  $\mathbf{U}$  takes values 0 or 1 depending on the fact whether the component is affiliated to the cluster/source or not.

In practice however, each component contains the interference of all sources to a certain degree. This fact is better expressed by a fuzzy clustering technique where the elements of  $\mathbf{U}$  take values from the close range [0,1]. This type of affiliation is called the *Fuzzy Partition*. The utilization of a fuzzy clustering technique within T-ABCD was proposed by the author of the thesis and is described in Chapter 5.



**The reconstruction** aims at transforming the clustered components into source responses on microphones.

First, the observation subspace is reconstructed according to the affiliation of the components to the  $k$ th cluster via

$$\widehat{\mathbf{S}}_k = \widehat{\mathbf{W}}^{-1} \text{diag}\{\mathbf{\Lambda}_{k1}, \dots, \mathbf{\Lambda}_{k(mL)}\} \widehat{\mathbf{W}} \tilde{\mathbf{X}}, \quad (4.5)$$

where the coefficients  $\mathbf{\Lambda}_{kj}$  are suitable positive weights which are computed from the values contained in the *hard partition* matrix  $\mathbf{U}$ .

When a perfect separation is assumed [4], then the **binary weights** are set to

$$\mathbf{\Lambda}_{kj} = \mathbf{U}_{kj}, \quad (4.6)$$

i.e. the weights of components affiliated with the source are equal to 1 and the weights of the non-affiliated components are equal to 0.

In practice, the separation is rarely perfect, e.g. due to an insufficient demixing filter length  $L$ . Here, the idea of **fuzzy weights** [17] appears to be suitable. Each component is affiliated with each of the sources to a certain degree. The clusters are still determined based on the SAHN algorithm, but the non-negative weights are determined via an ad-hoc formula

$$\mathbf{\Lambda}_{kj} = \left( \frac{\sum_{i \in K_k, i \neq j} \mathbf{D}_{ij}}{\sum_{i \notin K_k, i \neq j} \mathbf{D}_{ij}} \right)^\alpha, \quad (4.7)$$

where  $K_k$  contains the indices of components affiliated with the  $k$ th cluster and  $\alpha$  is a free parameter which determines the "fuzziness" of the weights. With growing parameter  $\alpha$  the fuzziness of the weights diminishes.

Subsequently, the estimates of source responses on microphones are computed. The estimate of the response of the  $k$ th source on the  $i$ th microphone is acquired by

$$\hat{s}_k^i(n) = \frac{1}{L} \sum_{\ell=1}^L (\widehat{\mathbf{S}}_k)_{(i-1)L+\ell, n+\ell-1}, \quad (4.8)$$

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

## 5. Author's Modifications of the T-ABCD Algorithm

The proposed changes [a5] consist in the utilization of a fuzzy clustering technique for the clustering of ICs (T-ABCD, Step 4). Since the separation performed by ICA is not perfect, the components cannot be simply clustered into groups corresponding to the original sources. It is more plausible to assume that the components are affiliated to a certain extent with all sources. These degrees of affiliation can be established through fuzzy clustering and utilized in the process of source reconstruction.

The approach proposed in this chapter replaces the SAHN [19] clustering algorithm, computing the *hard partition*, by the Relational Fuzzy C-Means method (RFCM), which produces the *fuzzy partition*.

## 5.1 RFCM applied in T-ABCD

Let a fuzzy partition be described by the affiliation matrix  $\mathbf{U}$  which is given by the set of equations

$$\begin{aligned} \mathbf{U}_{kj} &\in [0, 1] \\ \sum_{k=1}^c \mathbf{U}_{kj} &= 1 \\ \sum_{j=1}^K \mathbf{U}_{kj} &> 0, \end{aligned} \quad (5.1)$$

where  $K$  is the number of clustered objects/components and  $c$  is the number of clusters.

The output matrix of fuzzy affiliations  $\mathbf{U}$  is used in the reconstruction step within the modified T-ABCD algorithm. The reconstruction is still performed via (4.5), however the weights  $\mathbf{\Lambda}$  are determined according to one of the following approaches.

**The indirect application:** The fuzzy partition  $\mathbf{U}$  is transformed into a hard affiliation by assigning each component to the cluster with the highest affiliation degree. The weights  $\mathbf{\Lambda}$  are computed (as in original T-ABCD) via (4.7).

This approach exploits fuzzy clustering for the determination of the clusters only. The weights are computed according to a formula from the original version of T-ABCD, which is known to be suitable for the reconstruction of sources.

**The modified application:** The weights  $\mathbf{\Lambda}$  are defined by

$$\mathbf{\Lambda}_{kj} = \left( \frac{\mathbf{U}_{kj}}{1 - \mathbf{U}_{kj}} \right)^\alpha, \quad (5.2)$$

where  $\alpha$  denotes a positive adjustable parameter. We select  $\alpha = 2$ .

This approach fully exploits the information contained in the fuzzy partition  $\mathbf{U}$ . It is designed to emphasize the ICs with the highest affiliation to the cluster.

**The direct application:** The fuzzy affiliations  $\mathbf{U}_{ij}$  are applied directly as weights to the components, i.e.

$$\mathbf{\Lambda}_{kj} = (\mathbf{U}_{kj})^\alpha. \quad (5.3)$$

The experiments presented in Section 5.2 reveal that the direct application of the fuzzy affiliations  $\mathbf{U}$  leads to the best separation results from all considered possibilities.

## 5.2 Experiments

The experiment presented in this section compares the separation results of original T-ABCD with hard clustering to the results of T-ABCD modified by fuzzy clustering. Further experiments concerning the utilization of fuzzy clustering within the T-ABCD algorithm can be found in the reviewed thesis.

The data used in this experiment consisted of two distinct sounds (replayed on loudspeakers) recorded on eight microphones. Five times two sources were mixed together. There were six distinct sources available, including two male voices, two female voices, a typewriter sound and a Gaussian noise.

For the experiment, two of the eight microphone recordings were selected in order to obtain the determined mixtures. There are  $7 \cdot 8/2 = 28$  possibilities how to do so. When all five recorded source combinations are considered, it gives  $5 \cdot 7 \cdot 8/2 = 140$  different separation scenarios. The data were sampled at 16kHz.

The inner parameters of T-ABCD were as follows:  $L = 26$ ,  $N = 6000$ ,  $\alpha = 2$ ,  $f = 1.5$ . The reconstruction weights were within the original T-ABCD with SAHN computed via (4.7). The separation quality was evaluated via the BSS\_EVAL toolbox [14]. The results are averaged over all sources, microphones and scenarios.

The experiment compares various possibilities for the computation of the reconstruction weights  $\mathbf{A}$ . In Section 5.1 following variants were proposed: The indirect utilization of the fuzzy partition  $\mathbf{U}$ , the modified utilization via (5.2) and the direct application. The results, averaged over all scenarios and sources, are shown in Table 5.1.

	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 5.1: The separation results of the original T-ABCD with hard clustering and the modified version with implemented fuzzy clustering. Three formulas for the computation of the reconstruction weights  $\mathbf{A}$  from the fuzzy partition  $\mathbf{U}$  are considered.

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 both clustering techniques are similarly successful when clustering the ICs, because they determine the demixing 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 and low values of SAR. The direct listening to the estimated signals uncovers 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 slight improvement of the separation results. 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 non-stationary mixing is considered and the necessity of demixing filter updates in time arises.

## 6. The Online T-ABCD algorithm

This chapter concerns an adaptive algorithm for the blind separation of audio sources via ICA in the time-domain. The proposed algorithm is based on the T-ABCD method described in Section 4.2 and will be further denoted as *online T-ABCD* [a7].

Although the mixing process considered by online T-ABCD is potentially dynamic, e.g. due to moving sources, it is assumed that it changes slowly and may be considered stationary within short data segments. Therefore, during a mixture interval of length  $P$ , the classical convolutive mixing problem, which is described by (1.2), is considered.

The basic idea of online T-ABCD consists in the block-wise utilization of its batch counterpart on the stationary segments of the mixtures. The batch variant is modified in such a way that it respects the continuity of the mixing process by introducing memory into the algorithm. The demixing filters are updated via a single iteration of ICA and clustering in each segment. The adaptivity of the algorithm is controlled by learning parameters, which control e.g. the convergence speed of ICA.

The algorithm is shown to achieve good separation results with relatively short demixing filters ( $L \approx 30$ ). The online version outperforms its batch counterpart (in the terms of average output SIR) even when the source positions are fixed, because it is capable of better adaptation to the non-stationarity of the audio signals.

### 6.1 Description of the algorithm

The input of online T-ABCD consists of  $m$  mixture signals, each of length  $N$ . The proposed algorithm starts with the segmentation of these signals into overlapping blocks of length  $P$ , with the shift of  $T$  samples such that  $R = P/T$  is an integer. The overlap length of two consecutive blocks is thus  $P - T$ .

The  $I$ th block of the  $j$ th input signal will be denoted by

$$x_j^I(n) = x_j((I - 1) \cdot T + n), \quad n = 1, \dots, T. \quad (6.1)$$

The uppercase superscript  $I$  will be used to denote the data and quantities related to the  $I$ th block. A separation procedure described below is successively applied to blocks of input signals and outputs blocks of separated microphone responses of the source signals.

Like the batch variant, the online procedure first constructs the observation matrix (4.3). Subsequently, it (I) applies a simplified BGSEP algorithm to decompose the data matrix into its independent components and (II) uses a relational fuzzy clustering method to group the independent components to form independent subspaces that represent the separated sources. The third step (III) consists in the reconstruction of the separated signals in each block and averaging the signals in the overlapping windows.

### Step I: Independent Component Analysis via Simplified BGSEP algorithm

A simplified version of BGSEP is utilized as a separator within the online T-ABCD algorithm. The original BGSEP [11] is a second order statistics based ICA algorithm. It utilizes a general AJD scheme, incorporating weight matrices. BGSEP and its weight matrices are optimized for sources which have a Gaussian distribution and are piecewise stationary.

Let  $\mathbf{X}^I$  be the data matrix within the  $I$ th block of input signals, which is defined as

$$\mathbf{X}^I = \begin{bmatrix} x_1^I(1) & x_1^I(2) & \dots & \dots & x_1^I(P) \\ x_1^I(2) & x_1^I(3) & \dots & \dots & x_1^I(P+1) \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ x_1^I(L) & x_1^I(L+1) & \dots & \dots & x_1^I(P+L) \\ x_2^I(1) & x_2^I(2) & \dots & \dots & x_2^I(P) \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ x_m^I(L) & x_m^I(L+1) & \dots & \dots & x_m^I(P+L) \end{bmatrix}, \quad (6.2)$$

where  $L$  is a free integer parameter, corresponding to the desired demixing filter length.

The goal of Step I is to find such a demixing matrix  $\mathbf{W}^I$  that the rows of  $\mathbf{C}^I = \mathbf{W}^I \mathbf{X}^I$  are as independent as possible and correspond to the independent components (ICs) of  $\mathbf{X}^I$ .

Only a single iteration of the simplified BGSEP is performed in each segment of the mixed data. The algorithm is initialized by the demixing matrix  $\mathbf{W}^{I-1}$  estimated in the previous segment.

The speed of convergence of the simplified BGSEP is controlled by a learning parameter  $\beta_1^I \in [0.05, 1]$  which is based on the quality of clustering. It was found helpful to increase the speed when the clusters of ICs did not seem well separated in the previous block of data. Otherwise,  $\beta_1^I$  can be close to zero to maintain continuity. Details on the matter can be found in the reviewed thesis or [a7].

### Step II: Clustering of independent components

The ICs of  $\mathbf{X}^I$  are arbitrarily filtered versions of the original signals. There are groups of components which correspond to one of the sources and form thus independent subspaces within  $\mathbf{C}^I$ . These groups are determined via clustering of the ICs. Prior to clustering, a similarity measure among the components needs to be established, which becomes the basis for the clustering.

**Similarity of ICs** is quantified via the generalized cross-correlation coefficients known as GCC-PHAT [18]. These coefficient are invariant to magnitude spectrum and depend on phase spectrum only. This is suitable for quantification of similarity of ICs, because the magnitude spectra of ICs are random.

Let  $C_i^I(k)$  and  $C_j^I(k)$  denote the Fourier transform of the  $i$ th and  $j$ th component, respectively,  $i, j = 1, \dots, mL$ , and let  $k$  denote the frequency index. The GCC-PHAT coefficients of the components, denoted by  $g_{ij}^I(n)$ , are equal to the inverse Fourier transform of

$$G_{ij}^I(k) = \frac{C_i^I(k) \cdot C_j^I(k)^*}{|C_i^I(k)| \cdot |C_j^I(k)|}, \quad (6.3)$$

where  $*$  denotes the complex conjugation.

If the components correspond exactly to the same source, i.e. without any residual interference,  $g_{ij}^I(n)$  is equal to the delayed unit impulse function, where the delay cannot be greater than  $L$ . Hence, the similarity between the  $i$ th and  $j$ th component can be measured by  $\sum_{n=-L}^L |g_{ij}^I(n)|$ . Then, the matrix of mutual similarity  $\mathbf{D}$  can be computed, taking the information from previous segment into account, according to

$$\mathbf{D}_{ij}^I = \sum_{n=-L}^L |g_{ij}^I(n)| + \beta_2 \cdot \mathbf{D}_{ij}^{I-1}, \quad i, j = 1, \dots, mL, i \neq j, \quad (6.4)$$

where  $\beta_2$  is a learning parameter,  $0 \leq \beta_2 \leq 1$ .

**The clustering algorithm** utilized in online T-ABCD is the Relational Fuzzy C-Means algorithm (RFCM) from [20], which allows tracking of continual changes of the clusters. Single RFCM iteration is performed in each block. It outputs a  $d \times mL$  matrix of fuzzy affiliations of ICs to clusters/sources  $\mathbf{\Lambda}^I$ . The algorithm is initialized by previously estimated affiliations  $\mathbf{\Lambda}^{I-1}$ .

### Step III: Reconstruction

The contribution of ICs of the  $k$ th cluster to  $\mathbf{X}^I$  is given by matrix

$$\widehat{\mathbf{S}}_k^I = (\mathbf{W}^I)^{-1} \text{diag}[(\mathbf{\Lambda}_{k1}^I)^\alpha, \dots, (\mathbf{\Lambda}_{k,mL}^I)^\alpha] \mathbf{C}^I, \quad (6.5)$$

where  $\alpha$  is an adjustable positive parameter. This matrix has an analogous structure as  $\mathbf{X}^I$  in (6.2). In an ideal case  $\widehat{\mathbf{S}}_k^I$  contains delayed microphone responses of the  $k$ th estimated source only. The response of the  $k$ th source at the  $i$ th microphone is therefore estimated by

$$\hat{s}_k^{i,I}(n) = \frac{1}{L} \sum_{q=1}^L (\widehat{\mathbf{S}}_k^I)_{(i-1)L+q, n+q-1}, \quad (6.6)$$

where  $(\widehat{\mathbf{S}}_k^I)_{\alpha,\beta}$  is the  $\alpha\beta$ th element of the matrix  $\widehat{\mathbf{S}}_k^I$ .

Finally, the overall outputs of the online algorithm are synthesized by putting together the estimated blocks of separated signals. The overlapping parts are averaged using a windowing function, for example, the Hann window.

## 6.2 Experiments

Two experiments concerning the online T-ABCD algorithm are presented. Further experiments can be found in the reviewed thesis.

### Separation of sources with fixed positions

In this experiment, the online algorithm separates real-world stationary mixtures of speech signals. The positions of the sources and the microphones are fixed. The results of online T-ABCD are compared to the performance of its batch counterpart.

The experiment aims at showing that the online T-ABCD is able to adapt to the non-stationarity of the speech signals. This improves its average performance even when the mixing system is stationary. The experiment uses publicly available data from Hiroshi Sawada’s websites.<sup>1</sup>

The recordings of *four sources using four microphones* are considered. The original signals are utterances of the length 7 s sampled by 8 kHz. The reverberation time of the room is 130 ms. Omni-directional microphones are used to record the mixtures.

The online and batch T-ABCD were both applied with  $L = 30$ . The other parameters of the online method were set to  $P = 6144$ ,  $T = 512$ ,  $\beta_2 = 0.95$  and  $\alpha = 3$ . The separation results are evaluated block-by-block. The blocks are of the same size as in the online method .

The batch method estimates the demixing transform using a single block of the data only. This should be sufficient, since the mixing process is stationary. Table 6.1 summarizes the results averaged over all blocks, microphones and sources.

Table 6.1: Results for separation of sources at fixed positions.

	SIR[dB]	SDR[dB]	SAR[dB]
online T-ABCD	8.43	1.58	4.41
batch T-ABCD	5.01	0.08	5.79

Online T-ABCD achieves better results in terms of SIR and SDR than the batch algorithm. This verifies the fact that the online method is able to adapt the separating filters throughout the recordings respecting the nonstationarity of sources (*not* the changes in the mixing process which is stationary in this experiment). On the other hand, the time-invariant separation done by batch T-ABCD produces less artifacts as indicated by SAR.

### Separation of moving sources

This experiment shows the ability of online T-ABCD to separate the mixtures of moving sources. To this end, we consider data given in the task “Determined convo-

<sup>1</sup><http://www.kecl.ntt.co.jp/icl/signal/sawada/>

lutive mixtures under dynamic conditions” (Audio Signal Separation) in the SiSEC 2010 <sup>2</sup> evaluation campaign. The campaign was organized as part of the LVA/ICA 2010 <sup>3</sup> conference .

The data simulate dynamic conditions so that *the maximum of two of six speakers* located at fixed positions *around a stereo microphone array* are active at a time. The separating algorithm applied to the data thus needs to adapt to active speakers. The distances between microphones were 2 or 6 cm and the sampling rate was 16 kHz.

We compare the proposed online T-ABCD with the frequency-domain BSS method by Francesco Nesta et al [22]. The online method was applied with  $L = 30$ ,  $P = 6144$ ,  $T = 512$ ,  $\beta_2 = 0.95$  and  $\alpha = 4$ . The Nesta’s method uses FFT of the length 4096 samples with 75% overlap. As the method works offline, it was applied independently on disjoint blocks of 1 second where the maximum of two sources were active.

Table 6.2: Results for the separation of data simulating dynamic conditions

2 cm	SIR[dB]	SDR[dB]	SAR[dB]
o. T-ABCD	10.39	3.87	6.16
Nesta	11.21	4.59	6.54

6 cm	SIR[dB]	SDR[dB]	SAR[dB]
o. T-ABCD	9.25	1.94	4.43
Nesta	7.90	1.85	5.37

The proposed method appears to be slightly inferior to the frequency-domain method if the distance of the microphones is 2cm, but it achieves better results if the distance is 6cm. It can be concluded that online T-ABCD seems to outperform the frequency-domain algorithm in cases of larger microphone distances, where the spatial aliasing occurs.

## Computational demands

The presented experiments were performed on a computer with single core 2.6 Ghz processor with 2 GB RAM. Online T-ABCD was implemented in Matlab environment. The computational demands of the algorithm depend on the demixing filter length  $L$  and the sampling frequency of the mixtures. The mixture signals in Section 6.2 (separation of moving sources) were 3 minutes, 29 seconds long, sampled at 16kHz. Online T-ABCD separation lasted 14 minutes, 36 seconds ( $L = 30$ ). This separation task can be performed in real time, when the desired filter length  $L = 10$ . Mixtures of two sources sampled at lower sampling frequency 8 kHz can be separated in real-time when  $L = 18$ .

<sup>2</sup><http://sisec.wiki.irisa.fr/>

<sup>3</sup><http://lva2010.inria.fr/>



## 7. Conclusions and Future Work

### 7.1 Separation of non-stationary non-gaussian sources

The author contributed to the development of the Block EFICA algorithm, which is designed for the separation of non-stationary and non-gaussian sources.

The Block EFICA algorithm is based on the famous algorithm FastICA [6] and its efficient variant EFICA [10]. It consists of three consecutive steps. In the first step, the sources are pre-estimated via FastICA. Within the second step, the score functions of the sources are estimated. Subsequently, the obtained functions are used as non-linearities in the one-unit Block EFICA iterations, which refine the estimates of the sources. The algorithm is finished by the computation of constants which minimize the residual mean square error of the rows of the demixing matrix. The method is proven to be asymptotically efficient provided that the sources are block-wise stationary with a constant variance.

Extensive experiments show the ability of the proposed method to accurately estimate all types of sources, including real world signals which do not obey the model of the algorithm. Moreover, it is shown that the theoretical assumption of the constant variance of the sources does not deteriorate the method's practical performance, when signals with varying energy are separated.

### 7.2 Modifications of the T-ABCD algorithm

The blind separation of audio sources can be performed by methods based on ICA in the time-domain. An example of such a method is T-ABCD [4] proposed by Koldovský and Tichavský. The algorithm exhibits an advantageous modular structure which allows simple implementation of new features.

The separation performed by ICA within T-ABCD is usually not perfect and a single IC often contains interference from other sources. Therefore, in the clustering step, the component cannot be uniquely assigned to a single source, but should be affiliated with multiple sources to a certain degree. To reflect this ambiguity, the author proposed the utilization of the Relational Fuzzy C-Means algorithm [20] for clustering of the ICs.

The experiments suggest that RFCM slightly improves the separation performance of T-ABCD compared to the original hard clustering method and is less computationally demanding. Moreover, RFCM features a favorable iterative structure, which enables it for exploitation within the online version of T-ABCD.

### 7.3 Adaptive separation of audio signals

The original batch T-ABCD assumes that the mixing process does not change its inner parameters in time. In practice though, the mixing may be non-stationary, e.g. due to the movement of the sources.

The author proposed an online variant of the T-ABCD method which profits from the fact that the original T-ABCD is able to estimate the demixing filters using short data segments only. Within such segments, the mixing process can be considered stationary.

The proposed method is based on the block-wise application of the original T-ABCD, which is modified to respect the continuous changes in the parameters of the mixing system. The adaptation is performed via a single iteration of ICA/clustering algorithms in each of the blocks. The speed of adaptivity is controlled by learning parameters.

The experiments suggest that the algorithm is able to adapt to the changing position of the sources. Moreover, it is shown that the algorithm can adapt to the non-stationarity of speech signals as well. Regarding the computational burden; the separation can be performed in real time provided that the desired separation filter is reasonably long.

## 7.4 Future work

The intended future research is focused on the online or batch T-ABCD algorithms, since the remaining projects are concluded.

Online T-ABCD was subject to numerous tests. However, detailed testing regarding the influence of some of its inner parameters on the separation performance has not yet been performed. This concerns especially the length and the shift of the ICA segment along with the forgetting factor in (6.4). These parameter values were selected in an ad-hoc manner. Moreover, cases of instability, which deteriorate the performance of the algorithm, occasionally occur, especially when a shorter length of the ICA segment is used. These cases need to be revealed through testing and corrected. The intended testing will include the objective criteria such as SIR/SDR/SAR as well as listening tests.

Further development of the algorithm is directed towards simplification and acceleration. The most computationally demanding part of the method is the computation of similarities among ICs. In the current version, the similarities are based on the GCC-PHAT [18] coefficients. These can be replaced, for example, by spectral coherence. It is a standard tool in the analysis of time series, which measures the dependence between two time series. The computation of coherence is simpler than the computation of GCC-PHAT. The computational burden could be further lowered by the computation of the similarities directly from the ICA demixing matrix  $\widehat{\mathbf{W}}$ , if it is possible and usable.

The algorithm can be simplified by focusing on specific settings and scenarios. Recently, a paper [a10] concerning the implementation of online T-ABCD in C++ has been submitted. The paper focuses on the case of two sources recorded by two microphones. Here, some algebraic simplifications can be exploited. For example, the explicit formula for matrix inversion can be used. Furthermore, the affiliations of ICs to clusters need to be computed for the first cluster only, the second is just

complement to one.

For the general case when multiple microphones are used, the algebraic simplifications can be found as well. For example, the matrix inversion could be in certain cases replaced by its Taylor expansion approximation. However, this approach needs to be further examined: the application of simplifications may lead to the deterioration of the performance and stability of the algorithm.

Online T-ABCD is influenced by the ambiguity of the ordering within its ICA step. In most cases, its consequence is suppressed, due to a short shift of the ICA window and the continuous changes of the demixing matrix. However, if there appears an interval in the data where the separation via ICA cannot be performed and thus the continuity is disturbed, the estimated sources may switch channels. This problem arises for example, when the source positions are located behind each other and subsequently leave the cover. This situation could be probably solved by the speaker diarization process. It detects various speakers in the data and groups together the utterances spoken by the same speaker.

## Appendix: Automatic Classifiers for Medical Data from Doppler Unit

The appendix of the reviewed thesis discusses research which does not concern blind audio source separation. This research was conducted by the author of the thesis during the first year of his Ph.D. studies [a1, a2]. It deals with the automatic classification of medical data: signals originated in ultrasound measurement of blood flow in the arteries of the lower limbs are analyzed. The measurement is part of the screening examination for atherosclerosis.

The presented work is based on the results published in the author's master thesis. Unlike the original results introduced there, the author proposes a new set of features which includes attributes used for the diagnostics by human experts. The practical computation of the features is refined. The classifiers are re-trained and their extensive testing is performed. The re-design of the classifiers leads to an improvement of the recognition score by 5%.

### 8.1 Introduction and motivation

Atherosclerosis and diseases of the cardiovascular system pose a serious threat for the modern population. The manifestation of these diseases in human extremities is called the Peripheral Arterial Disease.

For fast noninvasive screening of PAD in diabetological and cardiological ambulances, ultrasonic Doppler devices have been used for a long time. These inexpensive units measure average blood flow velocity along with distant blood pressures on several typical places of the lower limb. The resulting signal is usually referred to as a *Doppler velocity waveform*. From the shapes of the waveforms the expert can detect PAD. Doppler units are notably cheaper than duplex scanners used in clinical medicine and are affordable in general medicine practice.

The following sections present a set of automatic classifiers which analyze the measured waveforms and affiliate them with classes which reflect the progress of PAD. These classifiers, along with the cheap Doppler probe, could help to identify the first phases of the PAD in general practice surgeries.

## 8.2 Automatic Classifier Design

The proposed classifiers are designed according to the principle of supervised learning. In order to train them, a large database of real-world Doppler waveforms had to be collected. In our case, the data come from the Regional Hospital in Liberec, where it has been acquired during the last few years. Prior to their use for research purposes, the data were made anonymous, all personal information of the patients was deleted.

### Measurement of Doppler waveforms

The waveforms represent the mean velocity of blood flow in the artery in a short time period. By default, the signals are measured along with blood pressures. Five standard locations on each leg are examined, i.e. there are 10 waveforms from one patient per one examination.

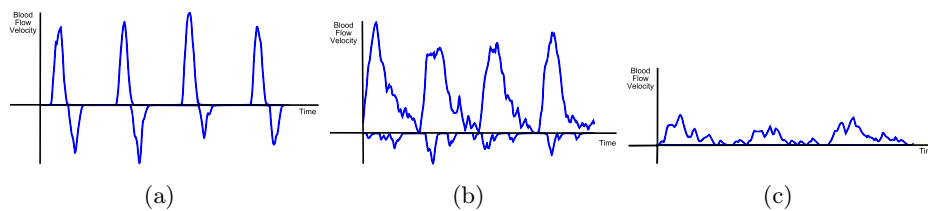


Figure 8.1: Representative waveforms of classes chosen for the automatic classification. Directional waveforms acquired in (a) artery with no PAD, (b) artery with stenotic diameter, (c) artery with total arterial occlusion.

### Classes

In accord with literature [23] and with the expert's opinion, four classes are chosen for the classification. Three of the classes reflect various degrees of artery occlusion. The remaining class contains waveforms which are deteriorated by some error encountered during measurement.

The classes are proposed as follows:

- **Normal course** – Waveforms acquired by the examination of arteries without PAD, e.g. Figure 8.1 (a).
- **Stenotic course** – Signals measured in arteries with a stenotic diameter, e.g. Figure 8.1 (b).

- **Occlusion**– Signals measured in arteries with a total arterial obstruction, e.g. Figure 8.1 (c).
- **Incorrect course** – This class covers four kinds of measurement errors which are commonly encountered during examination (for details see the reviewed thesis).

The expert of angiology classified part of the available database manually into designed classes before training of the classifier. This prior knowledge is used in the training process and also in the testing phase, when expert's opinion is compared with the results of the classifier.

### Features

During the design process, 18 features were considered as potentially useful for the classifier. Eventually, six most informative features was selected for the utilization.

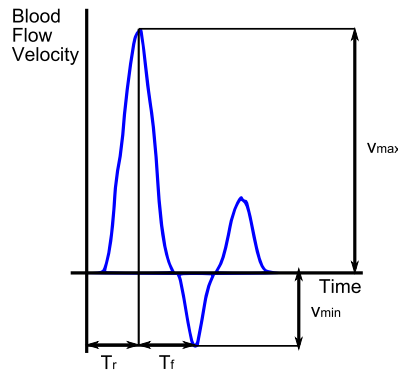


Figure 8.2: Proposed features of the Doppler waveforms

- **Brachial pressure index (BPI)** – the ratio of the patient's system blood pressure (measured on a. brachialis) and the distal pressure in the examined position on the lower limb.
- **Resistance (Pourcelot) index (RI)**

$$RI = \frac{v_{max} - v_{min}}{v_{max}}, \quad (8.1)$$

- **Deceleration (D)**

$$D = \frac{v_{max} - v_{min}}{T_f}, \quad (8.2)$$

- **Velocity-time index (VTI)**

$$VTI = \frac{v_{max} - v_{min}}{T_r + T_f}, \quad (8.3)$$

- **Frequency features F2 and F3** – The standard duration of measuring at one position is 5 seconds using the 100 Hz sampling frequency. The spectrum is calculated from the entire discrete signal via Fast Fourier Transform (FFT). The most of the energy in the spectra is concentrated up to one quarter of sampling frequency. These spectral coefficients are multiplied by eight triangle windows with half overlap in order to get 8 frequency features F1 to F8.

### Classifier types

Two basic types of classifiers were implemented during the design process: *The minimal distance classifier* and the *Bayesian classifier*. The classifiers analyze waveforms represented by a column feature vector of length  $M$ .

**The minimal distance classifier** (MDC) expresses each class by its best representative called prototype, which is a sample with the minimum distance to all others within given class. The implemented MDC classifier uses the Mahalanobis metrics, where the distance between two feature vectors  $\mathbf{x}, \mathbf{y}$  is given by

$$d(\mathbf{x}, \mathbf{y}) = \sqrt{(\mathbf{x} - \mathbf{y})' \Sigma^{-1} (\mathbf{x} - \mathbf{y})}, \quad (8.4)$$

where  $\Sigma$  is the sample covariance matrix of features within each class.

**The Bayesian classifier** (BC) represents each class  $C_i$ ,  $i = 1 \dots 4$  by a Gaussian probability density function (pdf) in the  $M$ -dimensional feature space. Its two parameters are the means and the variances. Since the classes have different occurrence rates, also the class prior probabilities are taken into account. The pdf is defined as follows:

$$P(\mathbf{x}|C_i) = \frac{1}{\sqrt{(2\pi)^M \cdot \det(\Sigma)}} \cdot \exp[-0.5(\mathbf{x} - \bar{\mathbf{x}})' \Sigma^{-1} (\mathbf{x} - \bar{\mathbf{x}})] \quad (8.5)$$

### Training and testing of the classifiers

For better modeling of the distributions of the feature vectors in the  $M$ -dimensional space, it is useful to split data in each class into clusters and represent each of them by a separate prototype or a separate pdf. In our case, the clusters are identified via the well known K-Means algorithm in combination with the Linde-Buzo-Gray algorithm (LBG) [24].

In order to train the classifiers and to make extensive tests, a large database of real-world Doppler waveforms was prepared by an expert. He classified data from 900 examinations manually. These were measured at 10 standard positions (5 on each leg), i.e. there were 9,000 sample waveforms available. Approximately 15% of all these signals were found incorrect. The reason was mostly setting of the amplification too high on a. femoralis, so that the signal was clipped. From the correctly measured ones, 47% were assigned to class Normal Course, 32% to class Stenosis and the rest 6% into class Occlusion. In the experiments, data from 720 randomly chosen examinations were used for training the classifier; the remaining data (of 180 subjects) were left for testing. In each individual test, the result of

the classifier was compared to the expert's decision. This was done for all test data and then the recognition score was calculated as ratio of correctly assigned to all available testing samples. The scores were calculated for each measured position and later averaged over all positions. To make the results more significant, the random database splitting into the training and testing part was repeated 5 times and the final scores were calculated as the means from the 5 tests. In other words, all the scores mentioned in the following section are averaged results from 9000 individual classifications (180 subjects x 10 positions x 5 repetitions).

### 8.3 Experiments

In Table 8.1, the comparison of the results from the two classifiers and their various settings is shown. The scores are based on correct decisions that include a) classification into a correct diagnostic class and b) correctly detected measurement error. It is evident that the best results were achieved by the Bayesian classifier with multi-modal pdfs and prior probabilities. The best score was 89%, i.e. the classifier and the expert agreed in 89% of the cases.

Table 8.1: Recognition scores for different classifier types and settings

Classifier	Setting	Score[%]
MDC	Mahalonobis - 1 cluster	81.98
MDC	Mahalonobis - Multiple clusters	83.65
BC	1-modal pdf without prior prob.	80.17
BC	1-modal pdf with prior prob.	84.46
BC	Multi-modal pdf without prior prob.	86.96
BC	Multi-modal pdf with prior prob.	89.19

### 8.4 Conclusions

Two types of automatic classifiers were designed for the Doppler waveforms. Large database of anonymous waveforms was compiled and subsequently used for training and testing of the classifiers. Based on the literature and the expert's opinion, the author introduced four diagnostic classes which reflect the progress of arterial occlusion. Eighteen features were proposed to describe the waveforms. The experiments revealed six the most informative ones. The testing of the classifiers led to the 89% agreement between the automatic classifier and the human expert.

# Bibliography

## List of cited papers

- [1] P. Comon, Ch. Jutten, *Handbook of Blind Source Separation: Independent Component Analysis and Applications*, Academic Press, ISBN: 978-0123747266, 2010.
- [2] A. Hyvärinen, J. Karhunen, E. Oja, *Independent component analysis*, Wiley-Interscience, New York, 2001.
- [3] I. T. Jolliffe, *Principal Component Analysis*, Springer, 2002.
- [4] Z. Koldovský and P. Tichavský, Time-Domain Blind Audio Source Separation Using Advanced ICA Methods, *Proc. Interspeech 2007*, pp. 846-849, August 2007.
- [5] A. J. Bell, T.J.Sejnowski, An information-maximization approach to blind separation and blind deconvolution, *Neural Computation*, Vol. 7, pp. 1129-1159, 1995.
- [6] A. Hyvärinen, E. Oja, A fast fixed-point algorithm for independent component analysis, *Neural Computation*, Vol. 9, No. 7, pp. 1483-1492, 1997.
- [7] J. F. Cardoso and A. Soulomiac, Blind Beamforming for non Gaussian signals, *IEEE Proceedings-F*, Vol. 140, No. 6, pp. 362-370, 1993.
- [8] A. Belouchrani, K. Abed-Meraim, J. F. Cardoso, and E.Moulines, "A blind source separation technique using secondorder statistics", *IEEE Transactions on Signal Processing*, Vol. 45, pp. 434-444, 1997.
- [9] D.-T. Pham, J. F. Cardoso, Blind separation of instantaneous mixtures of non stationary sources, *IEEE Transactions on Signal Processing*, Vol. 49, No. 9, pp. 1837-1848, 2001.
- [10] Z. Koldovský, P.Tichavský, E. Oja, Efficient Variant of Algorithm FastICA for Independent Component Analysis Attaining the Cramér-Rao Lower Bound, *IEEE Transactions on Neural Networks*, Vol. 17, No. 5, pp. 1265-1277, 2006.
- [11] 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.
- [12] D.-T. Pham, P.Garrat, C. Jutten, Separation of mixture of independent sources through a maximum likelihood approach. *Proc. EUSIPCO 1992*, pp. 771-774, 1992.



- [13] [Online] The Matlab *TM* implementation of the Extended EFICA algorithm: <http://itakura.ite.tul.cz/zbynek/eefica.htm>.
- [14] 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/](http://www.irisa.fr/metiss/bss_eval/)
- [15] S. Amari, S. C. Douglas, A. Cichocki, and H. H. Yang, Multichannel Blind Deconvolution and Equalization Using the Natural Gradient, *Proc. IEEE Workshop on Signal Processing Advances in Wireless Communications*, Paris, France, pp. 101-104, April 1997.
- [16] H. Bousbia-Salah, A. Belouchrani, and K. Abed-Meraim, Jacobi-like algorithm for blind signal separation of convolutive mixtures, *IEE Elec. Letters*, Vol. 37, No. 16, pp. 1049-1050, Aug. 2001.
- [17] Z. Koldovský, P. Tichavský, Time-domain Blind Audio Source Separation Using Advanced Component Clustering and Reconstruction, *Proc. HSCMA 2008*, pp. 216-219, 2008.
- [18] 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.
- [19] T. Hastie, R. Tibshirani, J. Friedman, *The Elements of Statistical Learning (2nd ed.)*, New York: Springer. pp. 520–528, ISBN 0-387-84857-6, 2009.
- [20] 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.
- [21] J. P. Rousseeuw, Silhouettes: a Graphical Aid to the Interpretation and Validation of Cluster Analysis, *Computational and Applied Mathematics*, Vol. 20, pp. 53-65, 1987.
- [22] F. Nesta, P. Svaizer, M. Omologo, A BSS method for short utterances by a recursive solution to the permutation problem, *Proc. SAM 2008*, Darmstadt, Germany, 2008.
- [23] Rozman, J. et al., Diagnostics of ischemic disease of lower limbs via Doppler measuring of blood flow velocity (in Czech), *Lékař a technika*, pp.76-78, pp.100-102, pp. 120-123, Praha, 1995.
- [24] Linde, Y., Buzo, A., Gray, R. M., An Algorithm for Vector Quantizer Design, *IEEE Transactions on Communications*, pp. 702-710, 1980.

### List of author's papers

- [a1] J. Málek: Bayesian classifier for medical data from doppler unit, *Acta Polytechnica*, Vol.46, No. 4, pp. 21-22, 2006.
- [a2] J. Málek, J.Nouza, T.Klimovič: Automatic classifiers for medical data from doppler unit, *Radioengineering*, Vol.16, No.2, pp. 62-66, 2007.
- [a3] J. Málek, Z. Koldovský, S. Hosseini, Y. Deville, A variant of algorithm EFICA with adaptive parametric score function estimator, *Proc. ECMS 2007*, Liberec, 2007.
- [a4] Z. Koldovský, J. Málek, P. Tichavský, Y. Deville, and S. Hosseini, "Extension of EFICA Algorithm for Blind Separation of Piecewise Stationary Non Gaussian Sources", *Proc. ICASSP 2008*, pp. 1913-1916, ISBN: 1-4244-1484-9, 2008.
- [a5] J. Málek, Z. Koldovský, J. Žďánský, J. Nouza, Enhancement of Noisy Speech Recordings using Blind Source Separation, *Proc. Interspeech 2008*, Brisbane, Austrálie, 2008.
- [a6] Z. Koldovský, J. Málek, P. Tichavský, Y. Deville, and S. Hosseini, "Blind Separation of Piecewise Stationary NonGaussian Sources", *Signal Processing*, Volume 89, Issue 12, pp. 2570-2584, ISSN 0165-1684, 2009.
- [a7] 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, ISBN: 978-3-642-15994-7, Springer, 2010.
- [a8] Z. Koldovský, P. Tichavský, and J. Málek, "Subband Blind Audio Source Separation Using a Time-Domain Algorithm and Tree-Structured QMF Filter Bank," in Latent Variable Analysis and Signal Separation, *Lecture Notes in Computer Science*, Volume 6365, pp. 25-32, ISBN: 978-3-642-15994-7, Springer, 2010.
- [a9] Z. Koldovský, P. Tichavský, and J. Málek, "Time-Domain Blind Audio Source Separation Method Producing Separating Filters of Generalized Feedforward Structure," in Latent Variable Analysis and Signal Separation, *Lecture Notes in Computer Science*, Volume 6365, pp. 17-24, ISBN: 978-3-642-15994-7, Springer, 2010.
- [a10] O. Hnilička, J. Málek, K. Paleček and Z. Koldovský, "A Fast C++ Implementation of Time-domain Blind Speech Separation Algorithm", to be published in *Proc. 20th Czech-German Workshop on Speech Processing*, Prague, 2010.



Ing. Jiří Málek

Blind Audio Source Separation via Independent Component Analysis  
*Doctoral Thesis Review*

Technical University of Liberec  
Faculty of Mechatronics, Informatics and Interdisciplinary Studies

30 pages  
10 copies

2010