Friedrich-Alexander University Erlangen-Nuremberg

Multimedia Communications and Signal Processing

Prof. Dr.-Ing Walter Kellermann

Research internship

Block-Online Implementation of Independent Vector Analysis

Stefan Wirler

September 2018

Supervisor: Thomas Haubner

Contents

List of Abbreviations

BSS	Blind Source Separation
STFT	Short-Time Fourier Transform
ICA	Independent Component Analysis
IVA	Independent Vector Analysis
AuxIVA	Auxiliary Independent Vector Analysis
SIR	Signal to Interference Ratio
RT	Reverberation Time

Chapter 1

Introduction

In a real-world acoustic environment, several conversations are happening at the same time. To distinguish between these conversations, a human is capable on focusing on one particular speaker. The other, non-wanted speakers, are suppressed to a certain level. This effect is called the "cocktail party effect". With the raising popularity of speech recognition systems, the need arises, to transfer this ability of human listener, into a technical system. An process to deal with this problem, is called Blind Source Separation (BSS). In most BSS systems speech is recorded with microphone-arrays, which consist of two or more microphones. The recorded signal is then processed by an algorithm, to separate desired signals from interfering signals. To process the recorded signals, several methods have been introduced in the past.

1.1 Model

For the task of BSS, the mixture of the speech sources (e.g a human speaker) and the demixing of these sources, can be described by an block diagram (see Fig. 1.1). The system consists of M sources and N microphones. The signals are described in the short-time fourier transform (STFT) domain, where k denotes the frequency bin and t the time-index. In the following the time-index t is omitted, due to convinience. The source signals can be written as,

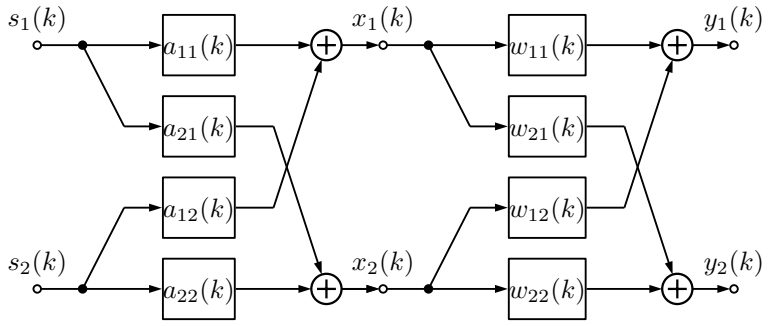

Figure 1.1: Mixing and demixing process

$$\mathbf{s}[\mathbf{k}] = [s_1(k), s_2(k), \ldots, s_M(k)]^T \tag{1.1}$$

where $s_M(k)$ describes the Mth source signal. The mixing process can be discribed as an instantaneous linear mixture of the sources, with the mixing matrix $\mathbf{A}(\mathbf{k})$. Therefore, the microphone signals $\mathbf{x}(k)$ can be written as

$$\mathbf{x}(\mathbf{k}) = \mathbf{A}(\mathbf{k})\mathbf{s}(\mathbf{k}) \tag{1.2}$$

.

with $\mathbf{x}(\mathbf{k}) = [x_1(k), x_2(k), \ldots, x_N(k)^T]$, where $x_N(k)$ is the N'th microphone signal. The goal of BSS is, to reverse the mixing of $\mathbf{A}(\mathbf{k})$, with an estimated separation matrix $\mathbf{W}(\mathbf{k}) = [\mathbf{w_1}(\mathbf{k}), \ldots, \mathbf{w_M}(\mathbf{k})]^H$, where $\mathbf{w_m}$ is a column vector and $(\cdot)^{\mathrm{H}}$ denotes the Hermitian transpose. The unmixed signal is then defined as

$$\mathbf{y}(\mathbf{k}) = \mathbf{W}(\mathbf{k})\mathbf{x}(\mathbf{k}) \tag{1.3}$$

The ideal separation is achieved, if $\mathbf{A}(\mathbf{k})\mathbf{W}(\mathbf{k}) = \mathbf{I}$. Where $\mathbf{I}$ is the identity matrix. The model described above, is a model which is used in many convulutive BSS approaches, e.g., Independent Component Analysis (ICA) [HO00].

1.2 Independent Vector Analysis

One way to estimate the source signals $\mathbf{s}(k)$ is by exploiting ICA. ICA aims at estimating $\mathbf{s}(k)$ with only the knowledge of the observed input $\mathbf{x}(k)$. The assumption is made, that the sources are statistically independent from each other. Thus, ICA aims at maximizing the statistical independence of the sources, to estimate the independent components. This can be achieved, by minimizing the mutual information between the sources with respect to the unmixing matrix [HO00]. With this minimization, a set of weight vector is found, which represent the unmixing matrix. The learning or the process of minimizing the mutual information is done for each frequency bin independently, with no consideration of interfrequency dependencies. Therefore, ICA is an effective method for separating sources at each frequency bin, but it suffers from permutations between frequency bins [Hir06]. This means, post-processing has to be used, to recover the source signal. Another approach deals with the permutation ambiguity problem by defining multivariate source priors. To achieve this Kim et al. proposed Independent Vector Analysis (IVA) [KALL07]. Instead of defining the source priors as independent priors, as in ICA, they assumed dependent source priors, which utilizes higher order dependencies between the frequency bins. In summary, IVA assumes dependency of the frequency bins within the same source and independency between the sources [HLLS10].

1.2.1 Objective Function of IVA

The estimation of the demixing matrix in IVA is done, by minimizing the following objective Function [Ono11].

$$J(\mathbf{W}) = \sum_{m=1}^{M} E\{G(\mathbf{y}_m)\} - \sum_{k=1}^{K} \log|\det W(k)| \tag{1.4}$$

where $E\{\}$ denotes the expectation operator. The contrast function $G(\mathbf{y}_m)$ holds following relationship

$$G(\mathbf{y}_n) = -\log(p(\mathbf{y}_m)), \tag{1.5}$$

where $p(\mathbf{y}_m)$ is a multivariate probability density function for each source. An often used contrast function, the spherical contrast function, is denoted as

$$G((\mathbf{y}_n) = G_R(r_m) \tag{1.6}$$

$$r_m = ||\mathbf{y}_m||_2 = \sqrt{\sum_{k=1}^{K} |y_m(k)|^2} \tag{1.7}$$

where $||\cdot||_2$ denotes the l_2-norm. This contrast function, is derived from the multivariate super-Gaussian source prior.

The minimization of the objective Function (Eq. 1.4) can be done by the natural gradient method [KALL07]. The minimization is done, by iteratively applying update rules, based on the natural gradient. But, this method suffers from a tradeoff between convergence speed and stability due to the step size. Therefore, in [Ono12] update rules, based on the auxiliary function technique, were proposed.

Chapter 2

Block-Online Auxiliary IVA

Auxiliary IVA (AuxIVA) uses the auxiliary function technique to improve the convergence speed and eliminate the step size. With the help of the auxiliary function technique a new objective function and update rules are defined [Ono12].

2.1 Auxiliary Function Technique

With the help of the auxiliary function technique the step-size optimization can be avoided. In the auxiliary function technique, the objective function is not directly minimized. Instead, an auxiliary function is defined, which will be minimized. The auxiliary function must satisfy

$$J(\boldsymbol{\theta}) = \min_{\tilde{\boldsymbol{\theta}}} Q(\boldsymbol{\theta}, \tilde{\boldsymbol{\theta}}) \tag{2.1}$$

with $J(\boldsymbol{\theta})$ is the objective function, $\boldsymbol{\theta}$ a parameter vector, $Q(\boldsymbol{\theta}, \tilde{\boldsymbol{\theta}})$ the auxiliary function and $\tilde{\boldsymbol{\theta}}$ an auxiliary variable. The variables are updated iteratively after following rule.

$$\tilde{\boldsymbol{\theta}}^{(i+1)} = \operatorname*{argmin}_{\tilde{\boldsymbol{\theta}}} Q(\boldsymbol{\theta}^{(i)}, \boldsymbol{\theta}) \tag{2.2}$$

$$\boldsymbol{\theta}^{(i+1)} = \operatorname*{argmin}_{\boldsymbol{\theta}} Q(\boldsymbol{\theta}, \boldsymbol{\theta}^{(i+1)}) \tag{2.3}$$

2.2 Objective Function and Update Rules

In the following, t denotes the time index of the STFT time blocks, due to the time dependency of the online separation process. The resulting objective function, determined with the auxiliary function technique, reads as

$$Q(\mathbf{W}, \mathbf{r}) = \frac{1}{2} \sum_{m=1}^{M} \sum_{k=1}^{K} \mathbf{w}_m^H(k) V_m(k) \mathbf{w}_m(k) - \sum_{k=1}^{K} \log |\det W(k)| + R. \tag{2.4}$$

$\mathbf{r}$ is a set of auxiliary variables (Eq. 2.5) r_m, with $m = (1, \ldots, M)$. The minimization of the objective function, is done by iteratively updating the auxiliary variables ($V_m(k)$, $r_m(k)$), and the demixing matrix until convergence is reached. The update steps in one iteration, over all frequency bins, for the auxiliary variables are denoted as

$$r_m = \sqrt{\sum_{k=1}^{K} |\mathbf{w}_m^H(k) \mathbf{x}(k,t)|^2}. \tag{2.5}$$

where $\mathbf{w}_m(k)$ is the weight vector for the m-th source and $\mathbf{x}(k)$ are the observations at the microphones. The auxiliary variable $r_m(t)$, is included in the weighted covariances $V_m(k)$, which reads as

$$V_m(k) = E\left\{ \frac{G'(r_m)}{r_m} \mathbf{x}(k) \mathbf{x}(k)^H \right\} \tag{2.6}$$

Besides Eq. **??**, another commonly used contrast function [Ono11], is described as

$$G_R(r) = m\text{logcosh}(Cr) \tag{2.7}$$

where C and m are positive constants.

The next step is the update and normalization of the weights of the demixing filter for each source. The update of the weights is denoted as

$$\mathbf{w}_m(k) = (\mathbf{W}(k) \mathbf{V}_m(k))^{-1} \mathbf{e}_m \tag{2.8}$$

with $\mathbf{e}_m$ as the unit vector, where m denotes the unit element. The last step of one

iteration is the normalization by following equation

$$\mathbf{w}_m(k) = \frac{\mathbf{w}_m(k)}{\sqrt{\mathbf{w}_n^H \mathbf{V}_m(k) \mathbf{w}_m(k)}} \tag{2.9}$$

2.3 Block-Online IVA

In the AuxIVA algorithm, there is only Eq. 2.6, which is dependent on all observations over time (expectation over time) [TOKS14]. Eq. 2.6 can be written as

$$V_m(k) = \frac{1}{N_t} \sum_{t=1}^{N_t} \left[\frac{G'(r_m(t))}{r_m(t)} \mathbf{x}(k,t) \mathbf{x}(k,t)^H \right] \tag{2.10}$$

Here, N_t denotes the number of total time blocks.

With this a online block-wise definition, of Eq. 2.10, can be written as

$$V_m(k,t) = \frac{1}{L_b} \sum_{\tau=t-L_b+1}^{t} \left[\frac{G'(r_m(\tau,t))}{r_m(\tau,t)} \mathbf{x}(k,t) \mathbf{x}(k,t)^H \right] \tag{2.11}$$

The weighted covariance $V_m(k,t)$ is the same as $V_m(k)$, which is calculated at time t. To obtain this weighted covariance the L_b past observations need to be stored. The calculation of $r_m(\tau,t)$ is equally to Eq. 2.5, but with the difference that only the observations at time-block τ are used. The difference of the other parameters in block-online AuxIVA is the time dependency.

For an adaptive algorithm an autoregressive calculation of $V_m(k,t)$ is used

$$V_m(k,t) = \alpha V_m(k,t-1) + (1-\alpha) \frac{1}{L_b} \sum_{\tau=t-L_b+1}^{t} \left[\frac{G'(r_m(\tau,t))}{r_m(\tau,t)} \mathbf{x}(k,t) \mathbf{x}(k,t)^H \right] \tag{2.12}$$

where α is the forgetting factor, which determines the change rate to the previously calculated V_m.

These described update rules, are computed at every time-step of the algorithm. Additionally, one or more iteration of the update rules can be applied in each time-step.

2.4 Scaling

Finally, one has to account for the scaling ambiguity of the demixing filters. Each frequency bin of the determined weighting, has an arbitrary scaling. To solve this problem the scaling is done, by applying the minimum distortion principle [Mat02]. In the block-online implementation, this has to be done at every time-step of the algorithm. The scaling can be written as

$$\mathbf{W}(k,t) \leftarrow \mathrm{diag}(\mathbf{Q}\mathbf{W}(k,t)^{-1})\mathbf{W}(k,t) \tag{2.13}$$

where $\mathbf{Q}$ is an permutation matrix. In most cases $\mathbf{Q} = \mathbf{I}$ ($\mathbf{I}$ is the identity matrix). After this step, the demixing filter can be applied to the observed signal at time t to recover the source signals.

Chapter 3

Evaluation

The evaluation of the algorithm was done, if not stated otherwise, with the parameters summarized in Table 3.1.

sampling rate (Hz)	16000
window	hamming
FFT window size	4096
FFT shift size	2048
number of iterations	2
contrast function $G_r(r)$	r

Table 3.1: Evaluation parameters

To evaluate the performance, the signal-to-interference ratio (SIR) is used [VGF06]. The SIR is defined as the ratio of the target energy to the interferer energy of the seperated signal, this can be written as

$$SIR := 10\log_{10}\frac{||s_{\text{target}}|\,|_2}{||e_{\text{interf}}|\,|_2}. \tag{3.1}$$

To measure the improvement, the difference of the output SIR, i.e., after separation and the input SIR, i.e., before separation, is calculated.
The SIR is computed by the toolbox described in [RMH+14].

3.1 Influence of α and Ł$_b$

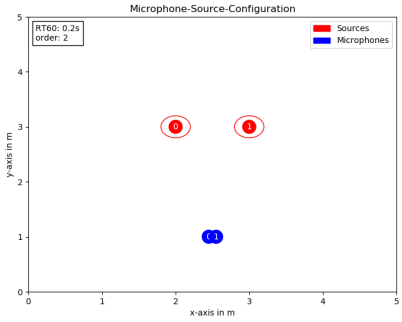

Figure 3.1: Scenario 1:Microphone and Source set-up

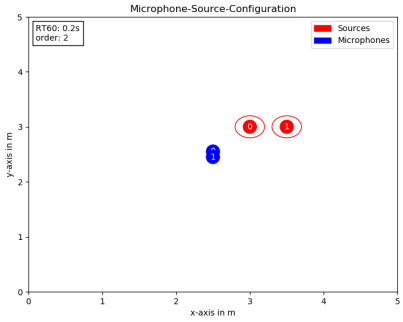

Figure 3.2: Scenario 2: Microphone and Source set-up

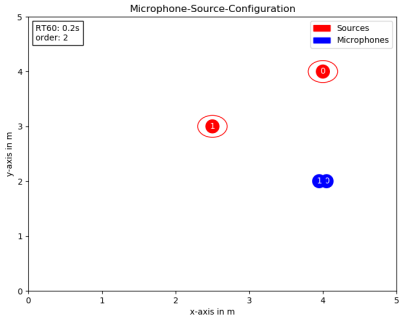

Figure 3.3: Scenario 3: Microphone and Source set-up

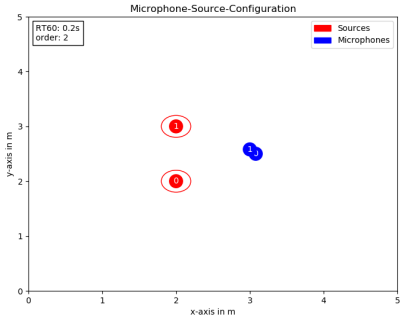

Figure 3.4: Scenario 4: Microphone and Source set-up

The first evaluation is done, for four different scenarios and a fixed block length. The SIR is evaluated for varying forgetting factor of $\alpha = \{0.7, 0.8, 0.9, 0.92, 0.94, 0.96\}$. The desired signal is represented by source 0 and the interferer by source 1. The room has a length of five meters, a width of five meters and a height of 2.5m. In the first scenario, depicted in Figure 3.1, the desired speaker is located at 2m x 3m (x-axis x y-axis) and the interfering or undesired speaker is located at 3m x 3m. The two microphones are located at 2.5m x 1m, parallel aligned to the x-axis, with a spacing of 10cm between the two microphones (spacing is chosen in every scenario). In the second scenario, shown in Figure 3.2, the desired speaker is located at 3m x 3m, the interfering at 3.5m x 3m and the microphone array at 2.5m x 2.5m, which is aligned parallel to the y-axis. For

the third scenario, the position of the desired speaker is set to 4m x 4m, the interferer to 2.5m x 3m and the microphone array at 4m x 2m, which can be seen in Figure 3.3. The position of the desired speaker for the last scenario, depicted in Figure 3.4, is set to 2m x 2m, the interfering speaker to 2m x 3m. The microphone array is positioned at 3m x 2.5m, with a tilt of 45°, w.r.t. to the x-axis. To measure the SIR, 6 different speakers (3 female speaker, 3 male speakers) are evaluated. The SIR is averaged over all 6 speakers and the four different scenarios. To observe the variation over time, the SIR is calculated over a period of two seconds, without overlap to the preceeding and succeeding period. The speech signals contain speech pauses, to simulate an real-world separation case. The latency, that is caused by the use of past blocks (at the beginning of separation) is equalized.

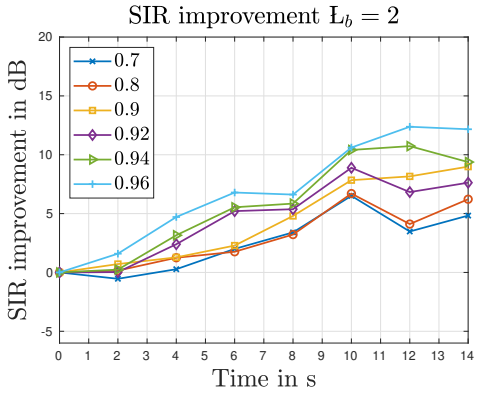

Figure 3.5: SIR improvement over time for varying α at $L_b = 2$

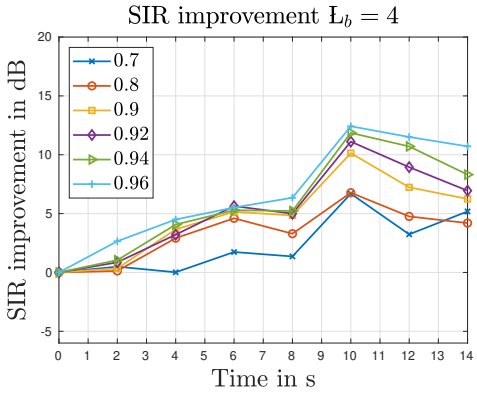

Figure 3.6: SIR improvement over time for varying α at $L_b = 4$

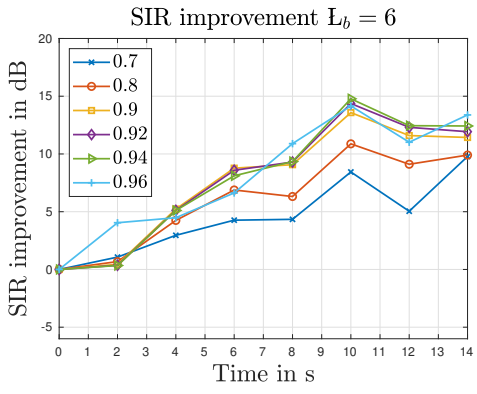

Figure 3.7: SIR improvement over time for varying α at $L_b = 6$

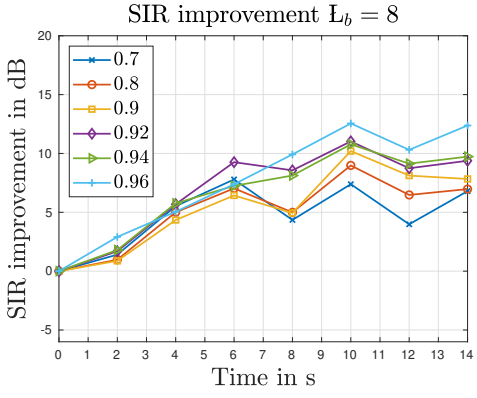

Figure 3.8: SIR improvement over time for varying α at $L_b = 8$

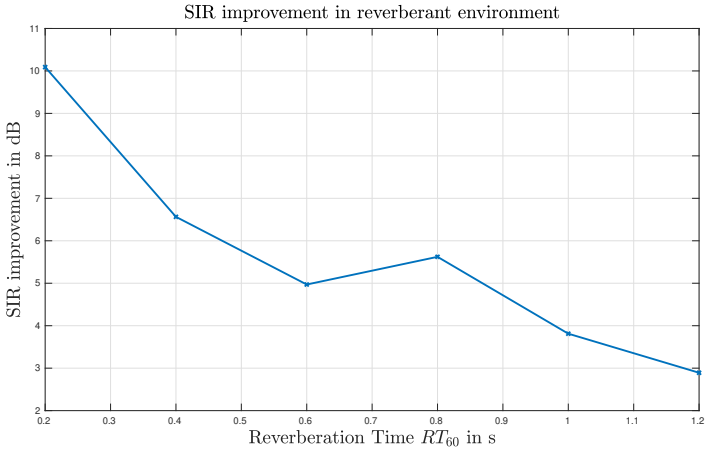

Figure 3.9: Influence of reverberation

For all evaluated block lengths, $\alpha = 0.96$ reaches the highest final SIR improvement. Up to a block length of $L_b = 8$, the algorithm converges faster and reaches higher SIR, for increasing block length. This effect can be explained by the additional information, that is induced by the higher number of used blocks. Hence, the estimate of the signal is better, and therefore better results can be reached.

3.2 Influence of Reverberation

In this section, the influence of reverberation is investigated. For the evaluation the block length is set to $L_b = 2$ and $\alpha = 0.96$. The overall SIR improvement, of the block-online separated signals, is calculated for 5 different combinations of speakers and then averaged. In Fig. 3.9, the SIR for $RT_{60} = \{0.2s, 0.4s, 0.6s, 0.8s, 1.0s, 1.2s\}$ is depicted. If the reverberation is stronger (higher reverberation time RT_{60}), the performance decreases. At low reverberation time the performance is good, but for $RT_{60} = 1.2s$ the overall SIR improvement drops down to approximately 3dB. Therefore, a highly reverberant environment leads to an unsatisfactory performance.

3.3 Different Contrast Function

In the following, the influence of an alternative contrast function on the performance is shown. The alternative contrast function is defined by Eq. 2.7. In Fig. 3.10 and Fig. 3.11 the difference of the SIR improvement of the spherical contrast function (see Eq. 1.6) and the alternative contrast function, are depicted($\Delta SIR = SIR_{alt} - SIR_{norm}$). The case of $L_b = 2$, depticted in Fig. 3.10, only a small difference can be observed, except at $\alpha = 0.9$. In the case of a high L_b and a smaller α the performance of the algorithm, when using the alternative contrast function, is worse compared to the spherical contrast function, as shown in Fig. 3.11.

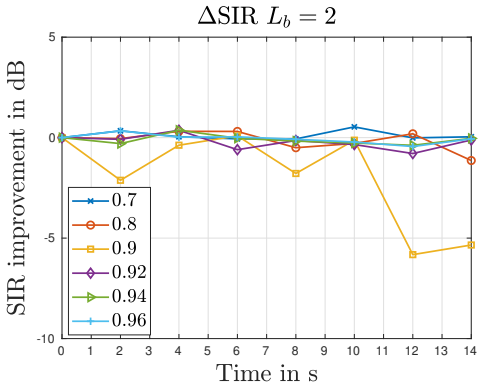

Figure 3.10: ΔSIR improvement for $L_b = 2$

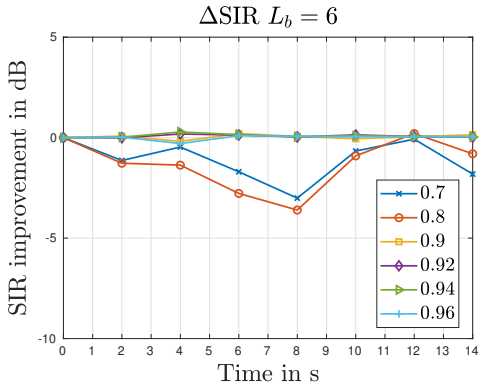

Figure 3.11: ΔSIR improvement for $L_b = 6$

3.4 Different Source Positions

For the evaluation of different source positions, two set-ups were investigated. In Fig. 3.13 the first set-up is shown. The position of the desired speaker (source 0), is changed by 30°, for every measurement, w.r.t. the center of the microphone array. The distance to the center of the microphone array stays constant for every angle. The second set-up, which consists of four different source position combinations, is depicted in Fig. 3.13 and evaluated for both sources. The block length is set to $L_b = 4$ and the forgetting factor to $\alpha = 0.94$. In Table 3.2 and 3.3, the results for the two set-ups are shown,

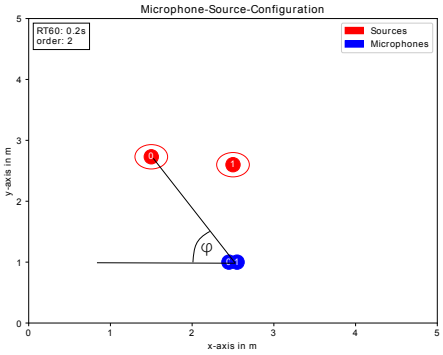

Figure 3.12: Set-Up 2: Microphone and Source positions

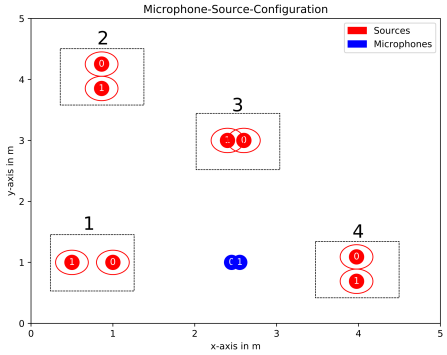

Figure 3.13: Set-Up 1: Microphone and Source positions

respectively. In the case of set-up 1, the overall SIR of the desired speaker, stays at about 10 to 12 dB if the desired speaker and the interferer have an sufficient distance. But for a more difficult separation case, the overall SIR drops around 6dB (see Table 3.2: $\varphi = 60°$), due to the closeness of the desired and the interfering source. In Table 3.3, both sources reach the best SIR improvement at position 1. The low SIR at position 3, source 1, is due to a low convergence speed.

φ	0°	30°	60°	90°	120°	150°	180°
SIR [dB]	12.05	11.70	5,84	0.66	5.84	11.19	12.05

Table 3.2: Interferer: distance to microphone = 1.6m, $\varphi_{interferer}$=90°

Source Position	1	2	3	4
Source 0 - SIR [dB]	8.36	8.49	7.65	6.59
Source 1 - SIR [dB]	8.73	6.80	2.90	5.97

Table 3.3: Overall SIR improvement for different Position combinations

3.5 Moving Sources

As this algorithm is an block-online implementation of the AuxIVA algorithm, the seperation performance for moving sources is of high interest. Therefore, the performance is investigated in this section, for two different situations. The positions of the moving source (source 0) and the interferer (source 1) is shown in Fig. 3.14 and in Fig. 3.15, respectively. The walking path is depicted as black line.

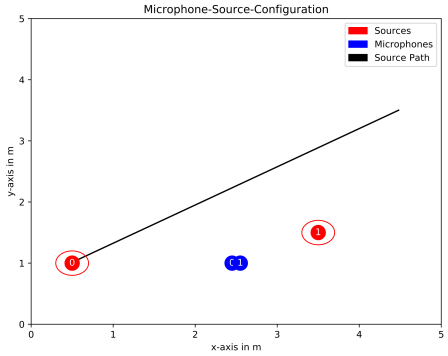

Figure 3.14: Situation 1: Microphone and source positions for moving sources

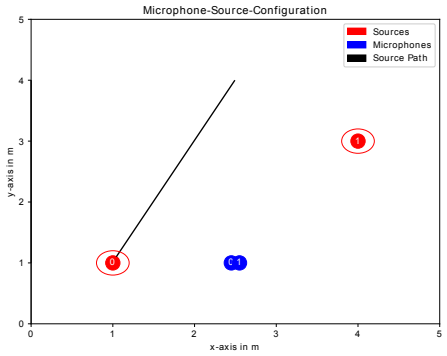

Figure 3.15: Situation 2: Microphone and source positions for moving sources

In situation 1, the movement of the source starts at 4s and ends at 9s, which corresponds to a normal human walking speed ($1.2\frac{m}{s}$). In Fig. 3.16 and Fig. 3.17 the SIR for $\alpha = 0.92$ and $\alpha = 0.96$ is depicted, with $L_b = \{2, 4, 6, 8\}$. The start and the end of the movement is indicated by the dashed line. The best performance is reached for $L_b = 8$, and the worst for $L_b = 2$. At first one would consider the use of more blocks as worse set-up for a moving source, due to the latency that is introduced by the use of past blocks. But in the case of normal walking speed of a source, the position difference of the blocks is not high enough to cause an degredation of the performance. It even improves the performance as seen in Fig. 3.16 and Fig. 3.17.

In Situation 2, another speaker pair is used. The walking of the moving source (source 0) is starting at 8 seconds and ends at 13 seconds. In Fig. 3.18 and Fig. 3.19, the effect of the use of a high number of blocks ($L_b = 30$) can be seen. The convergence is the highest of all used blocks lengths in the beginning. But after the start of the

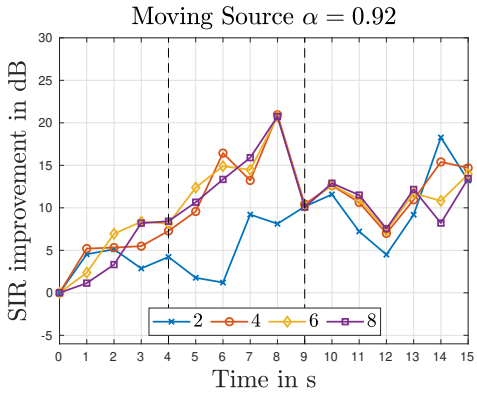

Figure 3.16: Situation 1: SIR improvement over time ($\alpha = 0.92$)

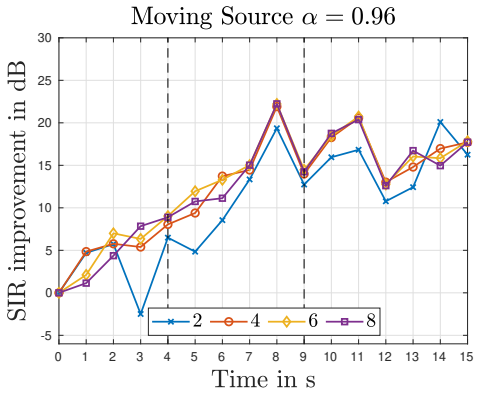

Figure 3.17: Situation 1: SIR improvement over time ($\alpha = 0.96$)

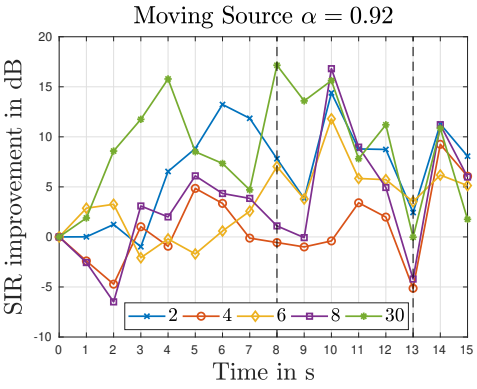

Figure 3.18: Situation 2: SIR improvement over time ($\alpha = 0.92$)

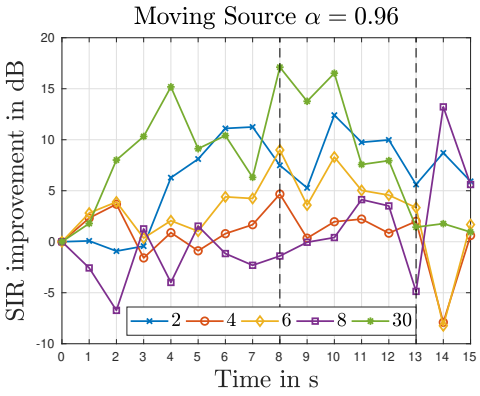

Figure 3.19: Situation 2: SIR improvement over time ($\alpha = 0.96$)

movement the SIR drops and reaches nearly 0dB at the end of the movement. This can already be explained by the use of the "wrong" information that is contained in the past blocks. Due to the latency, the algorithm can't follow the movement any more and will lead to an unsatisfactory result in terms of source separation. For a longer and a faster movement this effect will be increased. Therefore, a block length has to be chosen, which is suitable for the situation. Another drawback of the algorithm can be observed in Figure 3.18 and in Figure 3.19. For some combinations of α and Ł$_b$ the algorithm doesn't converges at all. In Figure 3.18 and in Figure 3.19 this effect is present for Ł$_b = 4$ and Ł$_b = \{4, 8\}$, respectively. This effect is present, when speech pauses of the signal, are dominant and non-overlapping. Therefore, in some cases, the

chosen block length leads to an insufficient estimation of the covariance matrix.

3.6 Back-Projection

The standard case for the back-projection permutation matrix is $\mathbf{Q} = \mathbf{I}$. This section evaluates 4 different permutation matrices (maximum number of permutation matrices for the case of two sources and two microphones) for different cases. The permutation matrix indicates, which weight of the demixing filter are used for the back-projection. The evaluated set-ups, can be seen in Fig. 3.20 and in Fig. 3.21.

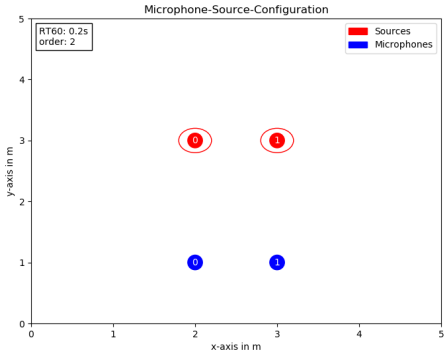

Figure 3.20: Set-up 1

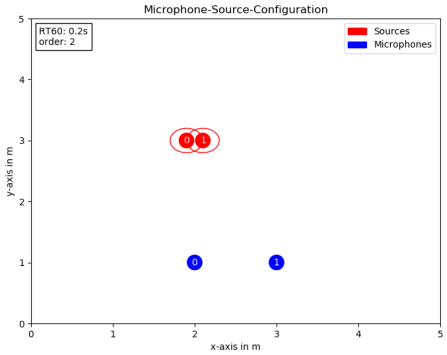

Figure 3.21: Set-up 2

$\mathbf{Q}$	$\begin{pmatrix}1&0\\0&1\end{pmatrix}$	$\begin{pmatrix}0&1\\1&0\end{pmatrix}$	$\begin{pmatrix}0&1\\0&1\end{pmatrix}$	$\begin{pmatrix}1&0\\1&0\end{pmatrix}$
Source 0 (SIR in dB)	19.17	18.75	19.17	18.75
Source 1 (SIR in dB)	22.42	22.46	22.46	22.42

Table 3.4: Set-up 1

In Table 3.4 and in Table 3.5, the results for different permutation matrices are listed. With the use of weights, which are more adequate for the scaling of the sources, an improvement of the overall SIR is achieved. Hence, better results are achieved with a variation of the permutation matrix, corresponding to the position of the sources. But this two cases, do not represent realistic scenarios. The positions of the sources

Q	$\left(\begin{smallmatrix}1&0\\0&1\end{smallmatrix}\right)$	$\left(\begin{smallmatrix}0&1\\1&0\end{smallmatrix}\right)$	$\left(\begin{smallmatrix}0&1\\0&1\end{smallmatrix}\right)$	$\left(\begin{smallmatrix}1&0\\1&0\end{smallmatrix}\right)$
Source 0 (SIR in dB)	16.66	17.46	17.46	16.66
Source 1 (SIR in dB)	18.14	16.44	18.41	16.44

Table 3.5: Set-up 2

Q	$\left(\begin{smallmatrix}1&0\\0&1\end{smallmatrix}\right)$	$\left(\begin{smallmatrix}0&1\\1&0\end{smallmatrix}\right)$	$\left(\begin{smallmatrix}0&1\\0&1\end{smallmatrix}\right)$	$\left(\begin{smallmatrix}1&0\\1&0\end{smallmatrix}\right)$
Source 0 (SIR in dB)	11.03	11.33	10.88	11.19
Source 1 (SIR in dB)	10.13	10.01	10.27	10.17

Table 3.6: Influence of permutation matrix - averaged over 50 random position combinations - First speaker combination

and the microphones were chosen, to show the influence of the permutation matrix. In order to reach a more realistic evaluation, 50 random generated position combinations of the sources and the microphone array were generated and the results were averaged over these combinations. This was done for three different speaker combinations.
The results of the three speaker combinations are shown in Table 3.6, 3.7 and 3.8, respectively. It can be seen, that in all three different cases, the maximum gain is smaller than 0.5dB. The highest gain is shown in Table 3.6, for source 1. The gain is at about 0.45dB. Therefore, in practical situations, where the position and the situations can alter, a use of a different permutation matrix has neither a postive and nor a negative effect. It was shown before, if the permutation matrix is adapted to the positions of the speakers, a gain can be reached. But this adaption would increase the computation time and for this reason it is not feasable yet.

Q	$\left(\begin{smallmatrix}1&0\\0&1\end{smallmatrix}\right)$	$\left(\begin{smallmatrix}0&1\\1&0\end{smallmatrix}\right)$	$\left(\begin{smallmatrix}0&1\\0&1\end{smallmatrix}\right)$	$\left(\begin{smallmatrix}1&0\\1&0\end{smallmatrix}\right)$
Source 0 (SIR in dB)	13.62	13.58	13.66	13.63
Source 1 (SIR in dB)	20.09	20.23	20.16	20.29

Table 3.7: Influence of permutation matrix - averaged over 50 random position combinations - Second speaker combination

Q	$\left(\begin{smallmatrix}1&0\\0&1\end{smallmatrix}\right)$	$\left(\begin{smallmatrix}0&1\\1&0\end{smallmatrix}\right)$	$\left(\begin{smallmatrix}0&1\\0&1\end{smallmatrix}\right)$	$\left(\begin{smallmatrix}1&0\\1&0\end{smallmatrix}\right)$
Source 0 (SIR in dB)	14.96	15.10	15.00	15.15
Source 1 (SIR in dB)	15.26	15.25	15.24	15.23

Table 3.8: Influence of permutation matrix - averaged over 50 random position combinations - Third speaker combination

Chapter 4

Summary

Auxiliary IVA is a promising method in the application of BSS. In this work, it was shown, that the use of numerous time blocks can lead to a better performance for an online source separation. With the use of time-blocks, the performance is increasing with the number of used blocks. It was shown that the performance in reverberant environment, as well as in other BSS algorithms, decreases with increasing reverberation. Further the ability, to separate moving sources, was investigated. In most cases, the use of a higher number of blocks leads to a better performance, but with the constraint, that the latency introduced by the past blocks is short enough to follow the alteration of the signal. There are cases for combinations of $Ł_b$ and α, where the separation failed. This was mostly observed for signals with many non-overlapping speech-pauses. A drawback of the algorithm is the optimization of the parameters $Ł_b$ and α, as there exists a wide variaty of combinations of these two parameters. Another constraint is the idle time at the beginning of the separation process. In practical applications the block length has to be chosen correspondingly to an idle time, that is acceptable for the separation situation. But after this idle time, the algorithm can perform separation in real-time. The next step, could be the further investigation of different cases for moving sources and different combinations of $Ł_b$ and α. Also a real-time implementation of the algorithm, could be focused.

Bibliography

[Hir06] Atsuo Hiroe. Solution of Permutation Problem in Frequency Domain ICA, Using Multivariate Probability Density Functions. In Justinian Rosca, Deniz Erdogmus, José C. Príncipe, and Simon Haykin, editors, *Independent Component Analysis and Blind Signal Separation*, pages 601–608, Berlin, Heidelberg, 2006. Springer Berlin Heidelberg.

[HLLS10] Jiucang Hao, Intae Lee, Te-Won Lee, and Terrence J. Sejnowski. Independent Vector Analysis for Source Separation Using a Mixture of Gaussians Prior. *Neural Computation*, 22(6):1646–1673, June 2010.

[HO00] A. Hyvärinen and E. Oja. Independent component analysis: algorithms and applications. *Neural Networks*, 13(4-5):411–430, June 2000.

[KALL07] T. Kim, H. T. Attias, S. Lee, and T. Lee. Blind Source Separation Exploiting Higher-Order Frequency Dependencies. *IEEE Transactions on Audio, Speech, and Language Processing*, 15(1):70–79, January 2007.

[Mat02] K. Matsuoka. Minimal distortion principle for blind source separation. In *Proceedings of the 41st SICE Annual Conference. SICE 2002.*, volume 4, pages 2138–2143 vol.4, August 2002.

[Ono11] N. Ono. Stable and fast update rules for independent vector analysis based on auxiliary function technique. In *2011 IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA)*, pages 189–192, October 2011.

[Ono12] Nobutaka Ono. Auxiliary-function-based independent vector analysis with power of vector-norm type weighting functions. In *Signal & Information Processing Association Annual Summit and Conference (APSIPA ASC), 2012 Asia-Pacific*, pages 1–4. IEEE, 2012.

[RMH+14] Colin Raffel, Brian McFee, Eric J Humphrey, Justin Salamon, Oriol Nieto, Dawen Liang, Daniel PW Ellis, and C Colin Raffel. mir_eval: A transparent implementation of common mir metrics. In *In Proceedings of the 15th International Society for Music Information Retrieval Conference, ISMIR*. Citeseer, 2014.

[TOKS14] T. Taniguchi, N. Ono, A. Kawamura, and S. Sagayama. An auxiliary-function approach to online independent vector analysis for real-time blind source separation. In *2014 4th Joint Workshop on Hands-free Speech Communication and Microphone Arrays (HSCMA)*, pages 107–111, May 2014.

[VGF06] E. Vincent, R. Gribonval, and C. Fevotte. Performance measurement in blind audio source separation. *IEEE Transactions on Audio, Speech and Language Processing*, 14(4):1462–1469, July 2006.