Circuits and Systems
Mekelweg 4,
2628 CD Delft
The Netherlands
http://ens.ewi.tudelft.nl/

CAS-2018-4517156

M.Sc. Thesis

Clock-Offset Invariant Beamforming in Wireless Acoustic Sensor Networks

Sofia-Eirini Kotti

Abstract

Clock synchronization among the nodes of a wireless acoustic sensor network (WASN) is a significant issue that affects the performance of multi-channel noise reduction schemes. Since independent sensors are utilized, each accompanied by its internal clock, clock offsets are inevitable, even if the mismatch in the sampling frequencies is negligible. In this thesis, clock offsets are mathematically modeled and the problem of multi-channel linear filtering for speech enhancement is addressed through signal subspace methods. For this purpose, the generalized eigenvalue decomposition (GEVD) of the cross-power spectral density (CPSD) matrices of the noise and target speech processes is capitalized. Beamformers based on this technique are proved to be invariant to sensor clock offsets when used in a blind manner, exploiting only network measurements. This result is confirmed through experiments in a simulated environment.

Clock-Offset Invariant Beamforming in Wireless Acoustic Sensor Networks

A Generalized Eigenvalue Decomposition Approach

Thesis

submitted in partial fulfillment of the
requirements for the degree of

Master of Science

in

Electrical Engineering

by

Sofia-Eirini Kotti
born in Mytilini, Greece

This work was performed in:

Circuits and Systems Group
Department of Microelectronics & Computer Engineering
Faculty of Electrical Engineering, Mathematics and Computer Science
Delft University of Technology

Delft University of Technology

Delft University of Technology
Department of
Microelectronics & Computer Engineering

The undersigned hereby certify that they have read and recommend to the Faculty of Electrical Engineering, Mathematics and Computer Science for acceptance a thesis entitled **"Clock-Offset Invariant Beamforming in Wireless Acoustic Sensor Networks"** by **Sofia-Eirini Kotti** in partial fulfillment of the requirements for the degree of **Master of Science**.

Dated: 12 July 2018

Chairman: _____
dr.ir. R. Heusdens

Advisor: _____
dr.ir. R. Heusdens

Committee Members: _____
dr.ir. R.C. Hendriks

_____
dr.ir. J.N. Driessen

_____
dr. J.A. Martinez Castaneda

Abstract

Clock synchronization among the nodes of a wireless acoustic sensor network (WASN) is a significant issue that affects the performance of multi-channel noise reduction schemes. Since independent sensors are utilized, each accompanied by its internal clock, clock offsets are inevitable, even if the mismatch in the sampling frequencies is negligible. In this thesis, clock offsets are mathematically modeled and the problem of multi-channel linear filtering for speech enhancement is addressed through signal subspace methods. For this purpose, the generalized eigenvalue decomposition (GEVD) of the cross-power spectral density (CPSD) matrices of the noise and target speech processes is capitalized. Beamformers based on this technique are proved to be invariant to sensor clock offsets when used in a blind manner, exploiting only network measurements. This result is confirmed through experiments in a simulated environment.

Acknowledgments

Approaching the end of my university path as a Master student, I look back and get a very warm feeling about my experience at TU Delft and the Netherlands. The current period denotes my transition to the "adult life" and I cannot help but feel overly grateful for the opportunities I have had recently.
Having these few lines to express my gratitude, I would like to thank all my professors at TU Delft for making my academic experience here so interesting and for adding fuel to my crave for learning and discovering. I would also like to thank my advisors, Richard and Richard, for their assistance during this thesis project and the eye-opening and fun meetings, and Antreas Koutrouvelis for always finding time for me and for his precious support and guidance throughout my time on the 17th floor. Kudos to everyone on this floor for sustaining such a cozy and inspiring atmosphere!
A big word of thanks goes to my classmates, together with whom we drilled our brains constantly, and all my friends at TU Delft and SoSalsa for the happy times, for the philosophical explorations, the endless library hours and the great dances.
Last but most, I would like to express my love and gratefulness to my family, my parents, my sisters and my brothers-in-law, who have been my pillar of strength, always questioning but in the end supporting my choices. And I could not leave out my cute baby nephews, to whose photos I resorted frequently during my work.

Thank you everyone!

Sofia-Eirini Kotti
Delft, The Netherlands
12 July 2018

Contents

List of Figures

List of Tables

1 Introduction

Portable personal devices, such as smartphones, tablets and laptops, are increasingly penetrating the business and private lives of people. These devices are equipped with multiple sensors for different functions and may support various kinds of wireless interfaces for data communication. From an acoustical point of view, these devices usually include embedded microphones and can therefore be used to form an ad hoc network with the goal of completing a speech signal processing task. An example can be a teleconferencing application, where the overlapping speech from multiple participants can result in poor intelligibility for the remote listener, especially in a reverberant room.

In systems like this, multi-microphone schemes, such as beamforming, can be used to enhance the noisy signal prior to broadcasting. Their major advantage is that they exploit spatial characteristics of the acoustic scenario, in addition to the spectral characteristics of the sources, and can distinguish among target speech sources and noise sources that have different positions.

A wide class of beamformers assume a fixed regularly arranged microphone array with accurately known microphone positions, and they usually also require knowledge of the direction of the desired sound source. Blind beamforming techniques are another category, which does not assume prior knowledge of the microphone and source positions.

Traditional array techniques are not always applicable to ad hoc distributed sensor networks, bevause they differ from a centralized array in several ways and that creates challenges for the processing of the recorded signals. For instance, they include limited-power processing units and not any dedicated audio hardware.

A critical component in these networks is the clock synchronization, as each sensor has its own clock and a common time system is needed for tasks of signal processing. This is often the cause of reduced beamforming performance. This problem is the main motivation for the work presented in this thesis, which focuses on how the performance of typical linear beamformers is adjusted under clock imprecisions.

1.1 Research Statement and Outline

To cover the above topic, in this thesis the following general research question is addressed:

What is the effect of sensor clock offsets on the noise reduction performance of linear filters and, particularly, filters based on signal subspace techniques?

The rest of the thesis is organized as follows. Chapter 2 presents some fundamental terms in the field of speech signal processing and introduces the mathematical model

for the rest of the work. In Chapter 3 the problem description is given, highlighting the importance of dealing with sensor clock offsets. In Chapter 4 signal subspace methods for speech enhancement are introduced and the form of optimal linear beamformers based on themis given. Chapter 5 presents the main contribution of this thesis, which is the incorporation of clock offsets into the system model and the study of how signal subspace-based beamformers perform when these are present. Chapter 6 provides simulation results that support the analysis in the preceding chapter. Finally, Chapter 7 gives a brief summary and critique of the findings and identifies areas for further research.

2 Preliminaries

This introductory chapter discusses some fundamental terms in the field of speech signal processing and presents the mathematical system model that the subsequent analysis is based on.

2.1 Background Theory

2.1.1 Speech Signal Processing

The focus of this thesis is *speech signals*. A speech signal is created by changes in air pressure and can be represented as a function of time $f(t)$, with f representing the air pressure at time t. Speech signals are captured by acoustic sensors (microphones) in an intricate manner. Microphones operate based on different physical principles but they all share the main function of converting the air pressure variations of a sound wave into alternating voltage fluctuations. A component used alongside microphones is the analog-to-digital (A/D) converter, which samples the analog electrical signal of the microphone at regular time intervals and converts it to a digital signal. This conversion is necessary since the devices (e.g., computers) where the signal analysis and processing take place are digital in nature and have finite precision available for the depiction of the signal values.

Speech signal processing entails all activities that concern the acquisition, storage, representation and manipulation of speech signals and the information they contain [1].

In principle, a signal can have any functional form and it is possible to produce signals, such as sound waves, with extraordinary richness and complexity. Signal analysis is important, as a means of extracting information, drawing conclusions and commencing the processing of the signals. This analysis can take place in different domains, principally the time and the frequency domains.

In the field of speech signal processing, most speech enhancement algorithms are performed in the frequency domain. Frequency analysis or *spectral analysis* is a powerful mathematical tool and has developed greatly since its advent. A signal can be converted between the time and frequency domains with a pair of mathematical operators called a *transform*. The transform relevant to this study is the *Fourier transform*, which converts a time function into a sum of sine waves of different frequencies, denoted *frequency components*, possibly infinite in number. After the processing in the Fourier domain, the inverse Fourier transform is used to reconstruct the signal into a time function.

The two most common Fourier representations for discete-time signals are the discrete-time Fourier transform (DTFT) for infinitely long data sequences, and the discrete Fourier transform (DFT) for finite-duration sequences. More information on the DTFT can be found in [1]. All information on the DFT necessary for this thesis can be found in Appendix A.2.3.

2.1.2 Fundamentals of Random Processes and Systems

Part of this thesis concerns wide-sense stationary (WSS) processes [2]. For WSS processes, the mean of the process $\mu_X(k) = \mathbb{E}[X(k)]$, where $\mathbb{E}(\cdot)$ is the mathematical expectation operator, is a constant, independent of time. In most analyses, it is assumed that the processes are zero-mean, meaning $\mu_X(k) = \mu_X = 0$. Moreover, the autocorrelation sequence $r_X(k,l) = \mathbb{E}[X(k)X(l)]$ depends only on the difference $k-l$, which is called the *lag*, and not the time itself. Permitting a slight abuse in notation, the zero argument is dropped and the autocorrelation is simply written as a function of the lag [2], as in $r_X(k,l) = r_X(k-l,0) \equiv r_X(k-l)$.

Applying the Fourier transform to the study of random processes is not straightforward, as they are collections of signals. That is why a different approach is adopted: The *power spectrum* or *power spectral density* (PSD) of a random WSS process $\{X_n\}$ is the Fourier transform of its autocorrelation sequence $r_X(k)$, i.e., $S_X(\omega) = \sum_{k=-\infty}^{\infty} r_X(k)\mathrm{e}^{-j\omega k}$.

The focus of this thesis is linear time-invariant (LTI) systems [2]. Such a system is fundamentally described by its *impulse response function*, which is its response to an impulse function, modeled as a Kronecker delta function for discrete-time systems. For an LTI system, knowledge of the impulse response function $h(n)$ suffices to describe the system, since the output to any arbitrary input $x(n)$ can be simply computed as

$$y(n) = \sum_{k=-\infty}^{\infty} x(k)h(n-k) = x(n) * h(n), \tag{2.1}$$

where $*$ stands for linear convolution.

This concept extends easily to multiple-input multiple-output systems. If an LTI system is excited by the inputs $x_p(n)$, $p = 1, \ldots, P$, which generate the outputs $y_q(n)$, $q = 1, \ldots, Q$, then the system outputs will be given by

$$y_q(n) = \sum_{p=1}^{P} x_p(n) * h_{p,q}(n),$$

where $h_{p,q}(n)$ is the impulse response connecting input $x_p(n)$ to output $y_q(n)$.

An LTI system is described in the Fourier domain by its *frequency response function* $H(\omega)$, which if the Fourier representation of $h(n)$. This function is obtained as

$$H(\omega) = \frac{Y(\omega)}{X(\omega)}$$

and is the linear mapping of the Fourier transform of the input $X(\omega)$ to that of the output $Y(\omega)$. The frequency response of a system is a special case of the *transfer function* of a system, which is defined in the Z-transform domain. More on this can be found in [3].

2.1.3 Time-Frequency Analysis

A signal in time domain may be regarded as a representation with perfect time resolution and no frequency information. On the other hand, the Fourier transform of a signal may be considered to have perfect spectral resolution but no time information; that is because, in principle, frequency analysis is conducted as an average over all time. As such, spectral analysis loses all chronological information and fails to convey when different events occur in the signal. This is not a problem for stationary signals, as their frequency components remain constant over time. However, it is a problem for non-stationary signals, such as speech and audio signals. In such a case it is important to investigate how the frequency content of a signal varies over time. What is more, this averaging over the complete history of the signal can be challenging, especially in real-time applications. That is why it would be useful to divide the signal into segments, so that the processing can begin before the entire signal has been received.

The above are reasons that led to the development of methods in the *time-frequency domain*. Time-frequency representations provide both temporal and spectral information at the same time. Thus, they are particularly practical for the study of signals containing time-varying frequency components. Given the number of different applications and some theoretical limitations that cannot be overcome, the problem of describing a signal in a joint time and frequency manner does not admit a unique answer.

The most typical time-frequency representation is obtained via the short-time Fourier transform (STFT). This is the method relevant to this study and it is analyzed in Appendix A.3. The STFT replaces the global Fourier analysis with a series of local analyses: the signal is localized by moving an observation window along the time axis, and applying the Fourier transform to obtain the frequency content of the signal for each position of the window. This transform provides a uniform resolution in time and frequency. In practice, the STFT is computed as

$$X_m(k) = \sum_{n=-N/2}^{N/2-1} x(n + mR) w_A(n) \mathrm{e}^{-j2\pi kn/N},$$

where $x(n)$ is the input signal, $w_A(n)$ is the analysis window function of length M, R is the window hop size in samples and $X_m(k)$ is the DFT of the windowed data centered around time mR, with m indicating the time frame. Often, it is $R < M$, in which case windows are overlapping. The number of frequency components has to be greater than the length of the windowed input data, which means that the DFT length should be $N \geq M$. It is typically $N = 2^j$, $j \in \mathbb{N}^+$, so as to accelerate the fast Fourier transform (FFT) algorithm.

The time signal is reconstructed using the Inverse STFT (inverse STFT (ISTFT)), which is the inverse DFT (IDFT) of this sum, possibly including a synthesis window $w_S(n)$

$$x(n) = \sum_{m=-\infty}^{\infty} \sum_{k=0}^{N-1} X_m(k) w_S(n - mR) \mathrm{e}^{-j2\pi k(n-mR)/N}.$$

2.2 Acoustic Concepts

In the field of acoustics, the room impulse response (RIR) function formally represents the sound transmission from a source to a particular receiving point in a room. It contains all information regarding the audible properties of the sound field in the specific acoustic scene. It covers all kinds of phenomena that a wave undergoes while it propagates, such as reflection, diffraction, refraction and scattering on obstacles.

This thesis follows the approach of geometrical room acoustics [4], which, admittedly, dictates a substantial simplification of the wave propagation laws. This simplification is ensured by adopting the notion of vanishingly small wavelengths. This assumption is justified when the dimensions of the room including all its details are large compared to the sound wavelength. In typical rooms, this holds for frequencies larger than 1000 Hz [4].

Geometrical room acoustics assumes that the sound sources are point sources and that the boundaries and surfaces in the room are plane and smooth. It considers reflections to be specular and reflection coefficients to be frequency-independent. In addition, it regards homogeneous media that do not cause any energy loss. Any typical wave effects other than reflection are neglected, since propagation in straight lines is its central premise. This hypothesis is invalid especially for low frequencies, where this method fails to accurately represent the sound propagation.

Obviously, this approach can only partially convey the acoustical phenomena occurring in a room. Nevertheless, it is of great importance because of its conceptual simplicity and the ease of computations it provides.

A simple description of the mechanism for reflections is the following: A spherical wave propagates away from the source in all directions. The sound that first reaches the receiver position corresponds to the wave component that has traveled directly from the source to the receiver and is called the direct sound. This holds provided, of course, that the direct path is not blocked by obstacles. This component is soon followed by others that have been reflected one or multiple times by boundaries or objects before reaching the receiver. Besides the difference in arrival time, normally these reflections are weaker than the direct sound. This is because the sound intensity is reduced as the area of the spherical wavefront increases (spherical distance attenuation) [5]. The reflections appear at first rather sporadically, later at higher density. In an idealized form, an RIR is composed of infinitely many reflections, if the surfaces do not cause any attenuation. It is more realistic to assume, though, that reflected wave components

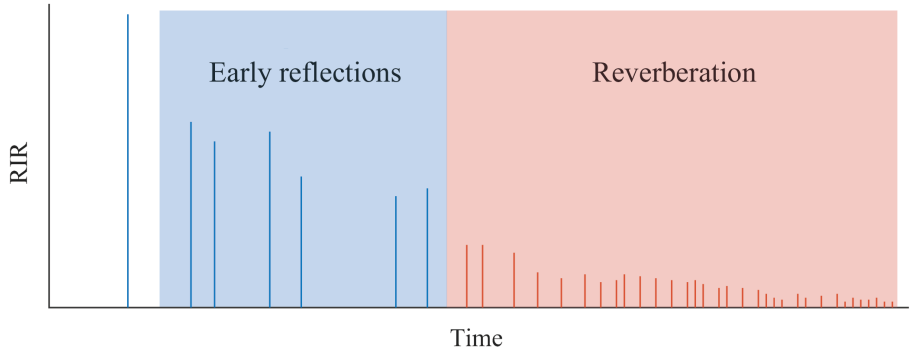


Figure 2.1: Illustration of a room impulse response function.

will continue to reach the receiver until all energy has been absorbed by the boundaries or objects.

If all absorption coefficients are frequency-independent, the observed signal at any position will be the superposition of (infinitely) many replicas of the source signal. The RIR is then

$$h(t) = \sum_n A_n \delta(t - t_n)$$

where A_n is the magnitude of the particular reflection, t_n is the respective traveling time for the wave and $\delta(t)$ is the Dirac delta function.

Although the reflection components of an RIR all admit the same physical description, they vary greatly from a subjective point of view. A reflection is not perceived as a separate event as long as its delay and amplitude compared to the direct sound do not exceed certain limits [4]. It only leads to perceiving the source sound as longer and louder. Thus, the "early reflections" give support to the source and improve the intelligibility of speech. The numerous weak reflections which reach the listener with longer delays merge into what humans perceive as reverberation and create a diffused noise environment, as they arrive at approximately the same level and time from multiple locations. These reflections may contribute, for example in a concert hall, to the warmth and brilliance of music. However, in acoustically non-specialized environments, such as a swimming pool, this late reverberation field can be very detrimental for speech intelligibility. This distinction of reflections is qualitatively illustrated in Fig. 2.1.

2.3 A General System Model

At this point, a general signal model will be introduced for this study. According to this model, N_s point sound sources which are static in space and (spatially and temporally) mutually uncorrelated are producing continuous time signals $s_n(t)$, $n = 1, \ldots, N_s$. These signals propagate in a reverberant room and impinge on an M-element microphone array. The signals captured by the microphones are further corrupted by

temporally white and spatially uncorrelated additive sensor noise with time signal $z_m(t)$, $m = 1, ..., M$, e.g., thermal noise. A homogeneous, lossless medium is assumed. For each source, the effect of the room reverberation and the microphone set-up on its signal is modeled as an LTI system. This system operates as a finite impulse response (FIR) filter on it. This clearly exposes that this model cannot account for the entirety of reflections in the room, but only for a number of them.

Under these assumptions, the continuous-time signal at the output of the m-th microphone is given by

$$y_m(t) = \sum_{n=1}^{N_s} h_{n,m}(t) * s_n(t) + z_m(t) \tag{2.2}$$

$$= \sum_{n=1}^{N_s} s'_{n,m}(t) + z_m(t), \tag{2.3}$$

$m = 1, \ldots, M$, where $h_{n,m}(t)$ is the acoustic RIR from the n-th source to the m-th microphone and $s'_{n,m}(t) = h_{n,m}(t) * s_n(t)$ is the clean signal originating from the n-th source, as observed at the m-th microphone. All previous signals are considered to be real and broadband. For now, it is also assumed that they are realizations of zero-mean WSS processes.

After sampling $y_m(t)$ and applying the DFT, Eqs. (2.2)–(2.3) are rewritten in the Fourier domain as

$$\begin{aligned} Y_m(f) &= \sum_{n=1}^{N_s} H_{n,m}(f) S_n(f) + Z_m(f) \\ &= \sum_{n=1}^{N_s} S'_{n,m}(f) + Z_m(f), \quad m = 1, 2, ..., M, \end{aligned}$$

where $Y_m(f)$, $H_{n,m}(f)$, $S_n(f)$, $S'_{n,m}(f) = H_{n,m}(f)S_n(f)$, and $Z_m(f)$ are the frequency-domain representations of $y_m(t)$, $h_{n,m}(t)$, $s_n(t)$, $s'_{n,m}(t)$, and $z_m(t)$, respectively. Note that the frequency components delivered by the DFT are discrete. Hence, variable f actually stands for f_k, where k is the frequency bin index. The subscript k is dropped for readability purposes.

Considering that the processing will take place separately for each frequency component, the frequency variable f will be omitted in the sequel, as in

$$\begin{aligned} Y_m &= \sum_{n=1}^{N_s} H_{n,m} S_n + Z_m \\ &= \sum_{n=1}^{N_s} S'_{n,m} + Z_m. \end{aligned}$$

If all M microphone signals in the frequency domain Y_m are stacked in a column vector the following representation is obtained

$$\begin{aligned}\mathbf{y} &= \sum_{n=1}^{N_s} \mathbf{h}_n S_n + \mathbf{z} \\ &= \sum_{n=1}^{N_s} \mathbf{s}'_n + \mathbf{z},\end{aligned} \tag{2.4}$$

where

$$\begin{aligned}\mathbf{y} &= [Y_1\ Y_2\ \cdots\ Y_M]^T, \\ \mathbf{h}_n &= [H_{n,1}\ H_{n,2}\ \cdots\ H_{n,M}]^T, \\ \mathbf{z} &= [Z_1\ Z_2\ \cdots\ Z_M]^T, \\ \mathbf{s}'_n &= \left[S'_{n,1}\ S'_{n,2}\ \cdots\ S'_{n,M}\right]^T \\ &= [H_{n,1}\ H_{n,2}\ \cdots\ H_{n,M}]^T S_n \\ &= \mathbf{h}_n S_n,\end{aligned} \tag{2.5}$$

and superscript ^{T} denotes the transpose of a matrix.

The elements $H_{n,m}$ in vector $\mathbf{h}_n$ of Eq. (2.5) are called the acoustic transfer functions (ATFs) and they are defined separately for each microphone m. As the frequency-domain equivalent of the RIR, they describe the frequency-dependent effects of both the environment (e.g., room reverberations) and the sensor setup on the n-th source signal [6]. The ATF vector $\mathbf{h}_n$ is indicative of the relative positions between the sound source n and the microphones. For this problem, the ATF vectors are considered time-invariant, since the sources and sensors are static in space and the room characteristics do not change.

From this point on, without loss of generality, the first microphone ($m = 1$) will be considered as the reference microphone. With this in mind, Eq. (2.4) can be rearranged in the following form

$$\mathbf{y} = \sum_{n=1}^{N_s} \mathbf{d}_n S'_{n,1} + \mathbf{z}, \tag{2.6}$$

where

$$\begin{aligned}\mathbf{d}_n &= [1\ D_{n,2}\ \cdots\ D_{n,M}]^T \\ &= \frac{1}{H_{n,1}}[H_{n,1}\ H_{n,2}\ \cdots\ H_{n,M}]^T \\ &= \left[1\ \frac{H_{n,2}}{H_{n,1}}\ \cdots\ \frac{H_{n,M}}{H_{n,1}}\right]^T \\ &= \frac{1}{H_{n,1}}\mathbf{h}_n,\end{aligned} \tag{2.7}$$

and $S'_{n,1} = H_{n,1}S_n$ is the clean speech DFT coefficient of the n-th source, as observed at the reference microphone $m = 1$.

The relative transfer function (RTF) is defined as the ratio between the ATFs of two sensors. Typically, in a multi-channel setting, one specific microphone is chosen as the unique reference, as was done above, and all ATFs are normalized with respect to it. In this case, the elements $D_{n,m}$ in vector $\mathbf{d}_n$ Eq. (2.7) represent the RTFs for the n-th source with respect to reference microphone $m = 1$. The RTF vector $\mathbf{d}_n$ is called the *steering vector*, it is, of course, frequency-dependent and, for this thesis, time-invariant.

It should be noted that for the general model discussed in this section and summarized in Eq. (2.6), no distinction was made between target sources and interfering sources.

2.4 A More Specific System Model

In this section, the model described in Section 2.3 will be particularized to the exact scenario that the rest of the thesis will study and the DFT analysis so far will be substituted by the STFT analysis.

Consider the model of Section 2.3 and suppose that, out of the N_s active sources, K are the desired sources to be preserved (target sources) and the rest are interfering sources. The signals are received by M sensors arranged in an arbitrary array. For the rest of this work, the effect of the noise sources will be clearly distinguished from that of the target sources. The continuous-time signal at the m-th microphone is

$$\begin{aligned} y_m(t) &= \sum_{n=1}^{K} h_{n,m}(t) * s_n(t) + \sum_{n=K+1}^{N_s} h_{n,m}(t) * s_n(t) + z_m(t) \\ &= x_m(t) + v_m(t), \end{aligned} \tag{2.8}$$

$m = 1, \ldots, M$, where $x_m(t) = \sum_{n=1}^{K} h_{n,m}(t) * s_n(t)$ is the received target signal at microphone m, and $v_m(t) = \sum_{n=K+1}^{N_s} h_{n,m}(t) * s_n(t) + n_m(t)$ is the total noise signal observed at the microphone, including the convolved interfering signals and the spatially and temporally uncorrelated sensor self-noise signal.

To obtain the STFT representation of the signal, the multiplicative transfer function (MTF) approximation will be exploited. This approximation assumes that the support of the RIRs is finite and sufficiently short compared to the duration of the STFT analysis window. If so, the convolution with an RIR in the time domain can be converted into a multiplication in the STFT domain. As the length of the analysis window increases, the MTF approximation becomes more accurate [7].

With this in mind, Eq. (2.8) is transformed into the STFT domain using a window of length N_{DFT}, as in

$$\begin{aligned} Y_m(k,l) &= \sum_{n=1}^{K} H_{n,m}(l)S_n(k,l) + \sum_{n=K+1}^{N_s} H_{n,m}(l)S_n(k,l) + N_m(k,l) \\ &= X_m(k,l) + V_m(k,l), \end{aligned}$$

where k is the frame number and l is the frequency bin index. Note that the LTI ATFs $H_{n,m}(k,l)$ do not change with time, therefore the frame number has been dropped.

Stacking the received microphone signals $Y_m(k,l)$, $m = 1,\ldots,M$, in a vector $\mathbf{y}$ and similarly for $X_m(k,l)$ and $V_m(k,l)$, leads to the following signal model

$$\mathbf{y}(k,l) = \mathbf{x}(k,l) + \mathbf{v}(k,l),$$

or, dropping the frame number k and the frequency bin index l from the notation, simply

$$\mathbf{y} = \sum_{n=1}^{K} \mathbf{d}_n S'_{n,1} + \mathbf{v}, \tag{2.9}$$

$$= \mathbf{x} + \mathbf{v}, \tag{2.10}$$

where $S'_{n,1}$ is the DFT coefficient of the n-th source at this frame and frequency bin, $\mathbf{d}_n$ is the steering vector containing the RTFs from the n-th source to the microphones and $\mathbf{y}$, $\mathbf{x}$ and $\mathbf{v}$ are of size $M \times 1$, .

For each particular frequency bin, it is assumed that the signals $\mathbf{y}$, $\mathbf{x}$ and $\mathbf{v}$ are realizations of the respective zero-mean WSS processes, the latter being denoted by the corresponding capital letter. It is further assumed that the received target and noise processes are spatially uncorrelated, meaning $\mathbb{E}[\mathbf{X}\mathbf{V}^H] = \mathbb{E}[\mathbf{V}\mathbf{X}^H] = \mathbf{0}_{M\times M}$, where the superscript ^{H} denotes the conjugate transpose operator and $\mathbf{0}_{M\times M}$ is a matrix of size $M \times M$ with all its elements equal to 0.

For each frequency bin, the cross-power spectral density (CPSD) matrix of the received process $\mathbf{Y}$ is given by

$$\mathbf{R}_\mathbf{Y} = \mathbb{E}[\mathbf{Y}\mathbf{Y}^H] = \mathbf{R}_\mathbf{X} + \mathbf{R}_\mathbf{V}, \tag{2.11}$$

where $\mathbf{R}_\mathbf{X} = \mathbb{E}[\mathbf{X}\mathbf{X}^H]$ and $\mathbf{R}_\mathbf{V} = \mathbb{E}[\mathbf{V}\mathbf{V}^H]$ are the CPSD matrices of $\mathbf{X}$ and $\mathbf{V}$, respectively. The size of these matrices is $M \times M$. Stemming from their definition, CPSD matrices are in general positive semidefinite matrices.

The term "cross" refers to the multichannel character of these complex PSD matrices. The CPSD matrix in the frequency domain is the equivalent of the correlation matrix in the time domain.

Using Eq. (2.9), it is obvious that the received target CPSD matrix is

$$\mathbf{R}_\mathbf{X} = \sum_{n=1}^{K} \sigma_{S,n}^2 \mathbf{d}_n \mathbf{d}_n^H, \tag{2.12}$$

where $\mathcal{S}_{n,1}$ refers to the n-th source target process and $\sigma_{S,n}^2 = \mathbb{E}[\mathcal{S}_{n,1}^2]$ is its variance. Obviously, it is $\text{rank}(\mathbf{R}_\mathbf{X}) = K$ since the MTF approximation is employed.

On the other hand, due to the sensor self-noise, it is $\text{rank}(\mathbf{R}_\mathbf{Y}) = \text{rank}(\mathbf{R}_\mathbf{V}) = M$.

2.4.1 Estimation of the Cross-Power Spectral Density Matrices

Access to the actual CPSD matrices of the processes is hardly ever possible. In practice, these matrices are estimated using the data available.

The received noise CPSD matrix $\mathbf{R_V}$ can be estimated as $\hat{\mathbf{R}}_\mathbf{V}$ during "noise-only" periods, whereas $\mathbf{R_Y}$ is estimated as $\hat{\mathbf{R}}_\mathbf{Y}$ during "speech and noise" periods. This necessitates the existence of a voice activity detector (VAD) [8] in the system. Direct access to the samples of the target signals is most often impossible. Therefore, it is typical to obtain the estimated received target CPSD as the difference of the two above or

$$\hat{\mathbf{R}}_\mathbf{X} = \hat{\mathbf{R}}_\mathbf{Y} - \hat{\mathbf{R}}_\mathbf{V}. \tag{2.13}$$

A customary approach is to estimate the CPSD matrices using temporal averaging. This is the chosen estimation method in this thesis: The STFT is performed on the respective signals and the average over a number of time frames is obtained. Thus, the estimation of $\mathbf{R_Y}$ is done using un unbiased sample covariance matrix [9], as

$$\hat{\mathbf{R}}_\mathbf{Y} = \frac{1}{N_Y} \sum_{n=1}^{N_Y} \mathbf{y}(n)\mathbf{y}^H(n)$$

where $\mathbf{y}(n)$ is the observed signal in time frame n and N_Y is the number of frames used.

The same method is used for the noise CPSD matrix, during "noise-only" periods, and

$$\hat{\mathbf{R}}_\mathbf{V} = \frac{1}{N_V} \sum_{n=1}^{N_V} \mathbf{v}(n)\mathbf{v}^H(n),$$

where $\mathbf{v}(n)$ is the noice signal in time frame n and N_V is the number of frames used.

This estimation process inevitably introduces inaccuracies, since the estimates reach the actual matrices only for infinite number of samples. This methodology corresponds to Welch's method of modified periodogram averaging. The reader is referred to [2] for an overview of non-parametric methods for spectrum estimation.

Problem Background and Related Work

3

This chapter sets the scene for the rest of the thesis by outlining the problem background; in particular, wireless acoustic sensor networks (WASNs) and their application to speech enhancement are discussed, along with the synchronization problems they face. An overview of related prior work that can be found in literature is given and, finally, the specific scenario for this thesis, where clock offsets are present, is described.

3.1 Problem Background

3.1.1 Beamforming for speech enhancement

Speech enhancement is concerned with improving some perceptual aspect of speech that has been degraded by noise [10], be that perception by humans or better decoding by systems. Speech enhancement algorithms aim at improving the performance of a system whose speech input is contaminated by noise, and this goal translates into improving the quality and intelligibility of degraded speech. Better speech quality is perceived as increased "pleasantness" of the speech signal and it is highly desirable, as it can reduce listener fatigue. Improved intelligibility means that a noisy signal becomes more comprehensible by the listener. All speech enhancement algorithms reduce the background noise to some extent and are, therefore, referred to as noise suppression, or reduction, algorithms.

The need for speech enhancement arises in a variety of situations in which the speech signal originates from a noisy location or propagates through a noisy communication channel. This noise may emanate from interfering sources, noisy system elements or environment reverberations. Examples are voice communication applications, such as cellular telephone systems, speech recognition systems, teleconferencing systems and hearing aids. Depending on the application, intelligibility may be weighed as more important than speech quality and vice versa.

Speech enhancement schemes are divided into single- and multi-microphone schemes. Multi-microphone speech enhancement algorithms, the topic of this study, use measurements from multiple microphones and exploit both temporal and spatial information. These algorithms are also often referred to as *acoustic beamforming methods*; beamforming is a signal processing technique that is used in sensor arrays with the purpose of directional signal transmission or reception. When used at the receiving end, the spatial samples collected by the array are processed with the aim of estimating the signal arriving from a specific direction, in the presence of noise and interfering sig-

nals. Thus, a beamformer performs spatial filtering to separate signals that overlap frequentially and temporally but originate from different locations [11].

As in other domains, in acoustic applications beamforming algorithms achieve spatial selectivity by relying on the concept that signals recorded by different microphones in a room include components that are delayed and scaled versions of each other. By appropriately adding these microphone signals, including delay and scaling compensation, signal amplification in a desired direction is accomplished. This thesis concerns spectral speech enhancement methods of linear beamforming, in which the noisy speech signal undergoes processing in the frequency domain.

Using the model $\mathbf{y} = \mathbf{x} + \mathbf{v}$ of Eq. (2.10), where $\mathbf{y}$ is the observed signal vector for the specific frequency bin, the linear beamforming process can be symbolized as a filtering operation with output

$$\hat{x}_{ref} = \mathbf{w}^H \mathbf{y},$$

where $\hat{x}_{ref}$ is the estimate of x_{ref}, the target speech DFT coefficient of this frequency bin and time frame at a reference microphone and

$$\mathbf{w} = \begin{bmatrix} w_1 & w_2 & \dots & w_M \end{bmatrix}^T$$

is a complex-valued filter with length M, equal to the number of microphones.

The objective is to extract the sum of the desired speech signals at the reference microphone while minimizing the contribution of the noise terms, with little or no distortion of the target signal [12]. In reality, a speech enhancement system faces an essential performance limitation: the compromise between speech distortion and noise reduction.

3.1.2 Wireless Acoustic Sensor Networks (WASNs)

Wireless Acoustic Sensor Networks (WASNs) fall under the general category of wireless sensor networks (WSNs), whose nodes consist of autonomous self-powered devices, equipped with sensing, processing and communicating facilities [13].

WASNs are designed for acoustic signal processing tasks; each node is equipped with one or more microphones and is connected to others via wireless links. A WASN allows to deploy a large number of microphones at various positions, and can be exploited in systems for hearing aids, speech communication, the acoustic monitoring of an environment [13], etc. WASNs can be deployed in wide areas, possibly close to target sources, which can provide a high input signal-to-noise ratio (SNR) for the nodes.

When used for speech enhancement purposes, the topology of a WASN may include a fusion center, with which all sensor nodes are able to communicate, either directly or indirectly via relay nodes [14]. This center is responsible for gathering all measured signals in the network and processing them using conventional centralized multi-channel noise reduction algorithms. It may coincide with one of the network devices. However, a set-up of this kind is not robust: the performance or even functionality of the network may collapse in case the fusion center or other nodes important for data propagation

in the network fail. In practice, WASN topologies are time-varying as nodes easily join or leave the network, for instance at the event of a defect or an empty battery. This raises scalability concerns for centralized algorithms. Other factors, such as power restrictions, the limited sensor communication range and privacy considerations, may render a fusion center undesirable in many applications [14].

Distributed speech enhancement algorithms have been developed in order to tackle the shortcomings of centralized processing, by emphasizing the data transfer in local neighborhoods and dividing the processing burden over multiple nodes. In such algorithms, each node collects observations from the neighboring nodes, broadcasts its own and processes these data locally. The goal is to obtain the same noise reduction performance as with centralized algorithms. This approach provides more scalable solutions for the network design of large WASNs. This is because local processing can reduce computational complexity and the required communication bandwidth [14], as transmissions of only the end result of local computations are required. Moreover, in case a node leaves the network, the remaining nodes can in general still perform the desired task, with the appropriate adaptation of the network topology. In particular, some of the proposed distributed multi-microphone algorithms for speech enhancement in WASNs can be found in [13, 15, 16].

The time-varying nature, together with the random connectivity due to the wireless communication range, gives WASNs a largely dynamic character, with unpredictable changes in network size and topology. This immediately points towards a design challenge for distributed algorithms: ensuring robustness against topology changes. This is complemented by the issue of user privacy: the WASN nodes may belong to other owners and not the users themselves. If so, then private data may become openly available, leading to serious privacy complications. Other problems arise from the limited per-node energy resources for computations and communication, the unknown geometry of the sensor array, the bandwidth usage and the fact that each node has only partial access to the network data.

Besides the above, a major problem in distributed signal processing is the fact that each device in the network has its own processor with an independent internal clock. Beamforming algorithms heavily depend on timing information, as hinted by Section 3.1.1. Thus, their performance will heavily degrade when these clocks are not synchronized [14]. This is the main topic of this thesis and is further explored in the next section.

3.2 Synchronization issues in WASNs

In distributed signal processing systems, every node samples the observed analog signals using its own A/D converter. This process is controlled by the individual clocks of the nodes. In general, clock synchronization is a critical component for WSNs, as most applications require the joint processing of time data belonging to different nodes. Synchronization provides a common time system for the operation of the nodes and enables functions such as data fusion, which is needed for extracting meaningful information.

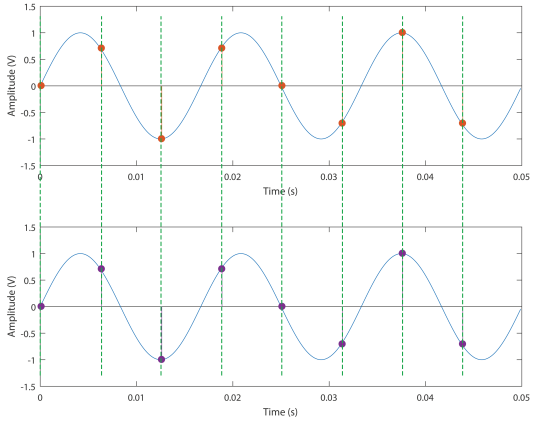
There are two aspects comprising the topic of clock synchronization in distributed sensor networks: the clock skew and the clock offset. These two are explained in detail below.

Time in most modern, inexpensive computers is derived from the oscillating frequency of a quartz crystal [17]. Due to environmental changes (in temperature, pressure, humidity, etc.) or minor manufacturing differences, variations in the crystal oscillation frequency in the order of 15–25 parts per million (ppm) compared to the nominal value are common [18]. A 32kHz oscillator commonly used for typical low-power sensor networks exhibits, in the worst case, a variation of 40 ppm, i.e., 40 μs per second [19] or 0.144 s per hour. The ratio of the actual sampling frequency over the nominal one is called the *absolute clock skew*. The ratio of the actual frequencies of two clocks is called the *relative clock skew*. It is obvious that the recorded signals of two sensors drift further apart as the frequency deviation accumulates over time. This problem is frequently encountered in literature as the sampling rate offset (SRO) problem.

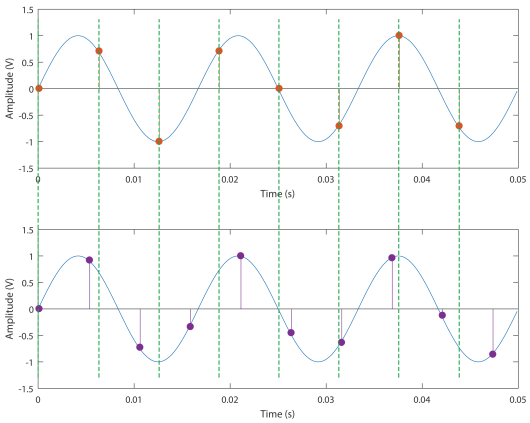
The *absolute clock offset* is defined as the difference between the local time of a node and the Coordinated Universal Time (UTC). In this work, the problem of the *relative clock offset* is considered, which refers to the difference between the local time and the time at a specified reference node clock inside the network. This offset depends heavily on the time stamping of the packets and is mainly affected by the non-determinism in the communication delay between nodes. A detailed description of sources of variability for this delay can be found in [20, 21]. The delay components can be generally categorized into two classes, deterministic and stochastic [18], and they mainly relate to the delay during the message assembly, encoding and decoding, and sending and receiving the entire length of a message over the channel. They may depend on the processed data load, the bandwidth and the packet size and are often collectively referred to as the "internal delay" of a sensor.

Fig. 3.1 shows a visual representation of the clock synchronization problem in a simplified setting. The setting is the following: Consider two microphones in the far field of a source equipped with perfectly synchronized to each other clocks. Imagine that the microphones have the exact same RIR and register precisely the same signal, which means that there is no self-noise at the sensors. In this case, the optimal beamformer is an adaptation of the Delay-and-Sum beamformer and it simply averages the two microphone signals. This configuration is depicted in Fig. 3.1a, whose top part shows the signal at the reference microphone and the bottom part at the second microphone. Continuous lines indicate continuous-time signals. The sampling frequency is arbitrary for this simplified example and the samples collected are designated by dots. The dashed vertical lines indicate the sampling moments for the first microphone. As easily verified, the second microphone samples at the exact same moments. As a result, employing the averaging beamformer will yield the original sampled waveform, as desired.

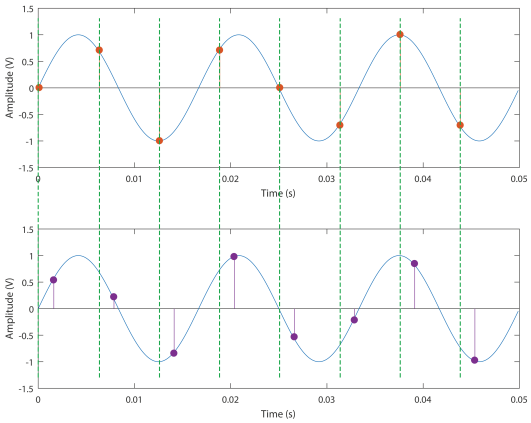
In Fig. 3.1b, the second microphone displays a clock skew with respect to the reference microphone. Particularly, it has a higher sampling frequency or skew greater than 1. Undoubtedly, the sampling moments of the two microphones do not coincide. It is clear that the horizontal time difference between samples collected by the two microphones with the same time tick increases. This means that the drift between samples iden-



(a) Perfectly synchronized clocks.



(b) Clock skew.



(c) Clock offset.

Figure 3.1: Illustration of clock synchronization issues.

tified by the same time tick grows. This phenomenon has a detrimental effect on the beamformer: averaging the two waveforms will by no means recover the initial signal, as the samples may be added constructively or destructively.

In Fig. 3.1c, the second microphone displays a clock offset with respect to the reference microphone, but no clock skew. Again, the sampling moments of the two microphones do not coincide, as indicated by the vertical lines. However, the horizontal time difference between respective samples collected by the microphones remains constant. This phenomenon also causes the beamformer to fail, as the averaging of samples does not reproduce the original signal.

Given the gravity of clock synchronization for signal processing in WSNs, several algorithms addressing this issue have been suggested [18, 22–25]. Despite this, most studies on multi-microphone speech enhancement in WASNs neglect the clock synchronization problem and are built on the implicit assumption that the independent clocks are perfectly synchronized, as for instance in [15]. If algorithms developed for other applications are to be used, it is important to check their applicability in practical scenarios of distributed speech enhancement systems, since they are based on different princi-

ples. Alternatively, it is of high value to investigate this issue specifically for WASNs, as recently done in [26, 27].

In a WASN with uniform hardware, ensuring equal sampling rates for the A/D converters is usually manageable [18] and sometimes even unnecessary if the oscillators are of sufficient quality, as evident in [26]. On the other hand, in non-uniform ad hoc WASNs with devices from different manufacturers, clock skew is most often present, and the resulting signal drift must be taken care of by dedicated synchronization algorithms.

Even if the mismatch in the sampling frequencies of independent devices are negligible, the origins of time are generally much different [28]. Additionally, more often than not, the sampling process does not launch simultaneously at all nodes, resulting in what is termed delay in sampling start (DISS) [26]. All in all, clock offsets are inevitable in WASNs and addressing them is essential for their operation. The problem of clock offsets in a WASN is the central matter of this thesis. The problem of clock skew is not examined.

3.3 Fixed Beamforming and Clock Offsets

In order to showcase the damaging effect of clock offsets on linear beamformers, the behavior of the minimum variance distortionless response (MVDR) filter was evaluated, when the actual location RTFs without clock offsets are used, but offsets are present in the network. For this, the case of one target source, three interferers, and five microphones with self-noise in a reverberant room was studied and the results are found in Fig. 3.2. Four microphones exhibit an equal clock offset with respect to the reference microphone, measured in samples, and the output SNR (see Section 6.1) is plotted. The input SNR (see Section 6.1) was set to $0dB$. The sampling frequency for all sensors is $16kHz$ and an STFT window of $32ms$ is used.

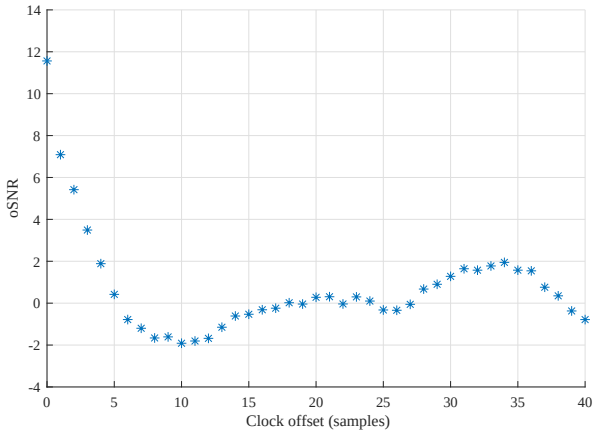


Figure 3.2: MVDR results when the location-based RTFs are used.

As the graph demonstrates, an offset of one sample for each microphone is enough to bring the noise reduction down by $4dB$. For larger offsets, this effect is stronger. This

confirms that beamforming based on knowledge of the devices location is not effective when clock offsets are present. The periodic elements the performance exhibits for larger offsets is related to the fact that clock offsets in the frequency domain affect the phase of the DFT coefficients in a periodic way. A closely-related proof for the clock skew problem and the delay-and-sum beamformer can be found in [14].

3.4 Related Work in Literature

In this section, a small review of prior work in literature in the field of clock synchronization for WASNs will be given.

In [29], time-of-arrival (TOA) measurements between sensors and sources, which include the internal delays of the sensors, are exploited to achieve distributed microphone localization. These delays are estimated using a structured total least squares (STLS) approach. The resulting optimization problem is solved by each node separately.

In [30], a two-microphone scenario is studied and the energy-based source separation technique of [31] is exploited to deal with both the clock skew and offset. This is a technique based on independent component analysis (ICA). For varying offset values and skew set to zero, the author concludes that the performance of this source separation technique is robust to a clock misalignment smaller than 12.5 ms. It should be noted that this maximum value for the clock misalignment is well within the limits of one frame, supposing it is set at between 20 and 30 ms. From then on, the paper focuses on estimating the clock skew and it is found that this sound source separation technique is less sensitive to drift errors than relevant TDOA-based ICA techniques.

In [28], a method for jointly estimating the time origins of different devices and localizing microphones and sources is proposed, using only recorded signals and assuming identical sampling rates. For this, initially a coarse alignment is performed according to the displacement that shows the maximum cross-correlation for the channel signals. Then, by calculating the cross-correlation frame by frame, single-source frames are identified and time differences between any channels for each source are obtained, which will include the differences in the time origins. Finally, an objective function is defined by the square errors of these differences and minimized.

The authors in [27] focus on the clock skew problem. They observe that the increasing signal drift is observed as a phase drift of the coherence between the signals and they use it in a weighted least-squares framework to estimate the skew. The above takes place assuming that the node with the reference clock acts as a central processor in the WASN.

In [32], an unsupervised method for estimating the clock skew and offset in an ad hoc microphone array is developed, meaning a method that does not require associating the clock time of a sensor to the absolute time. The writers highlight that unsupervised methods cannot deliver a precise estimate of the recording start offset without prior information. However, they claim that, in a blind scenario of array signal processing, a rough compensation of the recording start offset is sufficient and they obtain it by

using the time shift with the maximum correlation. The authors argue that, assuming the drifting time difference of signals is constant within each time frame, the effect of the sampling frequency mismatch can be compensated in the STFT domain by a linear phase shift. By considering motionless sources with stationary amplitudes, the observation will be stationary when drift does not occur. Thus, a likelihood function evaluating the stationarity in the STFT domain is used to calculate the drift compensation needed. All in all, the suggested method delivers accurate compensation of the drift and rough compensation of the recording start offset.

Many algorithms for distributed synchronization employ a series of time message transmissions. In [33], a two-stage procedure is used in a WASN. First, a two-way message exchange protocol together with a Kalman filter is used to estimate the clock frequency and time differences between pairs of nodes. In the second step, network-wide synchronization is achieved through a gossiping algorithm which estimates the average clock frequency and time of the nodes. These estimates are dealt with as frequency and time of a virtual master clock, to which the clocks of the sensors are adjusted.

Finally, in [26] the authors develop an algorithm based on wideband correlation processing. This algorithm provides accurate estimates for the SRO, without the assumption of a constant SRO during the observation period. However, as demonstrated, it does not deliver a reliable estimation of the DISS between the nodes. Nevertheless, the authors remark that under the assumption that the biased estimates provided are close enough the real DISS values, the bias effect can be absorbed into the estimation of the acoustic impulse responses [34].

3.5 Problem Formulation

The setting for this thesis is an ad hoc uniform WASN consisting of sensor nodes arbitrarily distributed in a reverberant room. Each sensor node has its own internal clock and contains a single embedded microphone. The devices are connected over a wireless network, and a central processor is available, which collects all network data and performs the multi-channel processing of the microphone recordings.

It is assumed that the clock frequency is precisely the same across microphones. However, clock offsets are present, which incorporate the differences in the moments the sampling process begins at each node. One node is chosen as reference and the rest exhibit a clock offset, either positive or negative, with respect to it.

The formulation of the problem at hand has two parts: firstly, modeling the clock offsets mathematically and, secondly, performing beamforming for noise reduction in the WASN while dealing with the clock offsets.

A Generalized Eigenvalue Decomposition Approach to Beamforming

4

In this chapter, the topic of beamforming for speech enhancement is addressed in more detail. In particular, the generalized eigenvalue decomposition (GEVD) framework to support signal subspace methods is described and applied to the system model of Chapter 2.

4.1 Introduction

The GEVD-based filters belong to the class of *signal subspace algorithms*. These algorithms are rooted primarily on linear algebra theory. More specifically, they are based on the principle that the clean signal might be confined to a subspace of the noisy Euclidean space [10]. That is why they take advantage of algebra methods to decompose the space of the observed signal into a subspace that is occupied primarily by the clean signal and a subspace occupied primarily by the noise signal. This decomposition can be done using the singular value decomposition (SVD) or the eigenvalue decomposition (EVD). Early work in this field was completed in [35] while more recent studies include [36, 37].

4.1.1 Eigenvalue Decomposition (EVD)

Let $\mathbf{A} \in \mathbb{C}^{n \times n}$ be a square matrix. A vector $\mathbf{z} \in \mathbb{C}^n$, $\mathbf{z} \neq \mathbf{0}$ and a scalar $\lambda \in \mathbb{C}$ fulfilling

$$\mathbf{A}\mathbf{z} = \lambda \mathbf{z} \tag{4.1}$$

are called an *eigenvector* and an *eigenvalue* of $\mathbf{A}$, respectively. The two of them together form an *eigenpair* of $\mathbf{A}$. To be more precise, $\mathbf{z}$ is a called a *right eigenvector* of $\mathbf{A}$ if it fulfills Eq. (4.1) and a *left eigenvector* if it fulfills

$$\mathbf{z}^H \mathbf{A} = \lambda \mathbf{z}^H. \tag{4.2}$$

By convention, unless otherwise stated, "eigenvector" means "right eigenvector".

It is easily seen that if $\mathbf{z}$ is an eigenvector tied to eigenvalue λ, this is also true for all vectors $\{\alpha\mathbf{z} : \alpha \in \mathbb{C}, \alpha \neq 0\}$. An eigenvector defines a 1-dimensional subspace that is invariant with respect to pre (or post-) multiplication by $\mathbf{A}$ [38]. A subspace $S \subseteq \mathbb{C}^n$ satisfying the property $\mathbf{x} \in S = \mathbf{A}\mathbf{x} \in S$ is said to be invariant for $\mathbf{A}$.

Every $n \times n$ matrix has exactly n eigenvalues, out of which some may occur with multiplicity greater than one. The eigenvectors connected to different eigenvalues are

linearly independent. For an eigenvalue with multiplicity greater than one, the associated eigenvectors are not necessarily linearly independent.

Let $\mathbf{V} \in \mathbb{C}^{n\times n}$ and $\mathbf{W} \in \mathbb{C}^{n\times n}$ be the matrices whose columns are, respectively, the right and left eigenvectors of $\mathbf{A}$ and let $\mathbf{\Lambda} = \text{diag}(\lambda_1, \ldots, \lambda_n)$ be a diagonal matrix holding the corresponding eigenvalues in its diagonal. Then, according to Eqs. (4.1) – (4.2), it holds that

$$\mathbf{AV} = \mathbf{V\Lambda} \tag{4.3}$$

$$\mathbf{W}^H\mathbf{A} = \mathbf{\Lambda W}^H. \tag{4.4}$$

If and only if matrix $\mathbf{A}$ has n linearly independent eigenvectors, then $\mathbf{V}$ and $\mathbf{W}$ are full-rank matrices, they are invertible, and Eqs. (4.3) – (4.4) can be rearranged into

$$\mathbf{A} = \mathbf{V\Lambda V}^{-1} \tag{4.5}$$

$$\mathbf{A} = \mathbf{W}^{-H}\mathbf{\Lambda W}. \tag{4.6}$$

Eqs. (4.5) – (4.6) give the eigenvalue decomposition (EVD), *eigendecomposition* or *spectral decomposition* of matrix $\mathbf{A}$. Matrices $\mathbf{V}$ and $\mathbf{W}$ are not unique, since any matrix of the form $\mathbf{VJ}$ and $\mathbf{WJ}$, where $\mathbf{J} \in \mathbb{C}^{n\times n}$ a diagonal matrix, will also give a spectral decomposition for $\mathbf{A}$.

If matrix $\mathbf{A}$ is Hermitian, then it definitely admits an eigendecomposition and its eigenvalues are all real. Moreover, a complete set of n orthonormal eigenvectors that form a basis in $\mathbb{C}^n$ can always be found for it. Hence, $\mathbf{V}^{-1} = \mathbf{V}^H$ so that $\mathbf{A} = \mathbf{V\Lambda V}^H$ and the left and right eigenvectors coincide.

4.2 Generalized Eigenvalue Decomposition (GEVD)

In the generalized Hermitian eigenvalue problem (GHEP) [39], non-trivial solutions to the problem

$$\mathbf{Au} = \lambda\mathbf{Bu} \tag{4.7}$$

$\mathbf{u} \in \mathbb{C}^n$ are sought, where matrices $\mathbf{A}, \mathbf{B} \in \mathbb{C}^{n\times n}$ are Hermitian, meaning $\mathbf{A}^H = \mathbf{A}$ and $\mathbf{B}^H = \mathbf{B}$. The pair $(\mathbf{A}, \mathbf{B})$ is called a *matrix pencil*. Here, the additional assumption is made that matrix $\mathbf{B}$ is positive definite, $\mathbf{B} \succ 0$, in which case $(\mathbf{A}, \mathbf{B})$ is called a *Hermitian definite matrix pencil*.

As proved in [38], for the Hermitian definite matrix pencil $(\mathbf{A}, \mathbf{B})$ there exists a non-singular $\mathbf{U} = (\mathbf{u}_1, \mathbf{u}_2, \ldots, \mathbf{u}_n)^T$, $\mathbf{u}_i \in \mathbb{C}^n$, such that

$$\mathbf{U}^H\mathbf{AU} = \text{diag}(a_1, \ldots, a_n) \tag{4.8}$$

$$\mathbf{U}^H\mathbf{BU} = \text{diag}(b_1, \ldots, b_n). \tag{4.9}$$

The vectors $\mathbf{u}_i$, $i = 1, \ldots, n$, are called the *generalized eigenvectors* that satisfy Eq. (4.7), thus $\mathbf{Au}_i = \lambda_i\mathbf{Bu}_i$, with $\lambda_i = a_i/b_i$ the *generalized eigenvalues*. As in the

EVD, the generalized eigenvectors are not unique. If $(\lambda_i, \mathbf{u}_i)$ is a generalized eigenpair, then so is every pair $(\lambda_i, \alpha\mathbf{u}_i)$ with $\alpha \in \mathbb{C}, \alpha \neq 0$.

Notice that normalizing the vectors $\mathbf{u}_i$ in a way such that $\mathbf{u}_i^H \mathbf{B} \mathbf{u}_i = 1$ will set $b_i = 1$ for all i. Then, Eqs. (4.8) – (4.9) can be rewritten as

$$\mathbf{U}^H \mathbf{A} \mathbf{U} = \mathbf{\Lambda} \tag{4.10}$$

$$\mathbf{U}^H \mathbf{B} \mathbf{U} = \mathbf{I}_n, \tag{4.11}$$

where $\mathbf{\Lambda} = \text{diag}(\lambda_1, \ldots, \lambda_n)$ and $\mathbf{I}_n$ is the $n \times n$ identity matrix.

From now on, the problem of the joint diagonalization of matrices $\mathbf{A}$ and $\mathbf{B}$ as in Eqs. (4.10) – (4.11) will be referred to as the GEVD problem, and only Hermitian definite pencils will be considered.

Since $\mathbf{B} \succ 0$, rearranging Eq. (4.7) demonstrates that $\mathbf{B}^{-1}\mathbf{A}\mathbf{u}_i = \lambda_i \mathbf{u}_i$, for $i = 1, \ldots, n$. The latter means that the GHEP problem for the matrix pencil $(\mathbf{A}, \mathbf{B})$ is equivalent to the eigenvalue problem for the matrix product $\mathbf{B}^{-1}\mathbf{A}$ and that the generalized eigenpairs $(\lambda_i, \mathbf{u}_i)$ are the right eigenpairs of $\mathbf{B}^{-1}\mathbf{A}$. It should be noted, however, that the matrix $\mathbf{B}^{-1}\mathbf{A}$ is not necessarily Hermitian; thus, $\mathbf{U}$ is in general not unitary, thus $\mathbf{U}^{-1} \neq \mathbf{U}^H$, and the vectors $\mathbf{u}_i$ do not constitute an orthogonal basis for $\mathbb{C}^n$. They, nevertheless, form a basis for $\mathbb{C}^n$. Note that the product $\mathbf{B}^{-1}\mathbf{A}$ is Hermitian if and only if $\mathbf{A}$ and $\mathbf{B}$ commute, or $\mathbf{AB} = \mathbf{BA}$.

However, it is true that

$$\mathbf{B}^{-1}\mathbf{A} = \mathbf{B}^{-1/2}\left(\mathbf{B}^{-1/2}\mathbf{A}\mathbf{B}^{-1/2}\right)\mathbf{B}^{1/2} = \mathbf{B}^{-1/2}\mathbf{S}\mathbf{B}^{1/2}, \tag{4.12}$$

with $\mathbf{S} = \mathbf{B}^{-1/2}\mathbf{A}\mathbf{B}^{-1/2}$, where $\mathbf{B}^{1/2}$ is the unique Hermitian square-root of $\mathbf{B}$. Eq. (4.12) reveals that $\mathbf{B}^{-1}\mathbf{1}A$ is similar to the Hermitian matrix $\mathbf{S}$ and, therefore, the two matrices share the same, real eigenvalues.

These n eigenvalues λ_i may be ordered decreasingly as $\lambda_1 \geq \lambda_2 \geq \ldots \geq \lambda_n$, and several eigenvalues may coincide, as in the standard eigenvalue problem. If, additionally, $\mathbf{A}$ is positive semidefinite, $\mathbf{A} \succeq 0$, it is $\lambda_n \geq 0$, whereas if both matrices $\mathbf{A}$ and $\mathbf{B}$ are positive definite, it is $\lambda_n > 0$, thus all eigenvalues are positive [39].

Rearranging Eqs. (4.10) – (4.11) reveals that

$$\mathbf{A} = \mathbf{Q}\mathbf{\Lambda}\mathbf{Q}^H \tag{4.13}$$

$$\mathbf{B} = \mathbf{Q}\mathbf{Q}^H, \tag{4.14}$$

where $\mathbf{Q} = \mathbf{U}^{-H} = (\mathbf{q}_1, \mathbf{q}_2, \ldots, \mathbf{q}_M)^T$, $\mathbf{q}_i \in \mathbb{C}^n$. Bearing in mind that

$$\mathbf{Q}^H\left(\mathbf{B}^{-1}\mathbf{A}\right) = \mathbf{Q}^H\left(\mathbf{Q}\mathbf{Q}^H\right)^{-1}\left(\mathbf{Q}\mathbf{\Lambda}\mathbf{Q}^H\right) = \mathbf{\Lambda}\mathbf{Q}^H,$$

it is obvious that $\mathbf{Q}$ contains the left eigenvectors of $\mathbf{B}^{-1}\mathbf{A}$ as columns.

The left and right eigenvectors, of course, do not coincide, but they are bi-orthogonal. This means that

$$\mathbf{q}_i^H \mathbf{u}_j = \delta_{ij}, \tag{4.15}$$

for all $i, j = 1, \ldots, n$, where δ_{ij} is the Kronecker delta function defined as

$$\delta_{ij} = \begin{cases} 1 & \text{if } i = j, \\ 0 & \text{if } i \neq j. \end{cases}$$

This stems from the fact that $\mathbf{Q}^H\mathbf{U} = \mathbf{U}^{-1}\mathbf{U} = \mathbf{I}_n$.

4.3 The GEVD for the System Model

In this section, the GEVD will be obtained for the system matrices introduced in Section 2.4. For this, the model $\mathbf{y} = \mathbf{x} + \mathbf{v}$ is considered, with $\mathbf{y}, \mathbf{x}, \mathbf{v}$ vectors of length M, where M is the number of microphones. It is assumed that all signals originating from the K target speech sources are incorporated in $\mathbf{x}$, whereas the effect of interfering sources is absorbed within $\mathbf{v}$.

Now, the theory of Section 4.2 will be applied to the matrix pencil $(\mathbf{R_X}, \mathbf{R_V})$, with $\mathbf{R_X}, \mathbf{R_V} \in \mathbb{C}^{M \times M}$, as defined in Section 2.4. Due to the sensor self-noise, the received noise CPSD matrix $\mathbf{R_V}$ is full-rank, therefore $\mathbf{R_V} \succ 0$. The received target CPSD matrix is generally $\mathbf{R_X} \succeq 0$. The rank of $\mathbf{R_X}$ is equal to the number of target sources, $\text{rank}(\mathbf{R_X}) = K$. The GEVD of $(\mathbf{R_X}, \mathbf{R_V})$, will give, according to Eqs. (4.10) – (4.11),

$$\mathbf{U}^H\mathbf{R_X}\mathbf{U} = \mathbf{\Lambda} \tag{4.16}$$

$$\mathbf{U}^H\mathbf{R_V}\mathbf{U} = \mathbf{I}_M, \tag{4.17}$$

where $\mathbf{U}$, $\mathbf{\Lambda}$ are of size $M \times M$ and $\mathbf{\Lambda} = \text{diag}(\lambda_1, \ldots, \lambda_M)$. It is $\lambda_i \geq 0$ for all i, as justified in Section 4.2.

The right generalized eigenpairs $(\lambda_i, \mathbf{u}_i)$ are, equivalently, the right eigenpairs of matrix product $\mathbf{R_V}^{-1}\mathbf{R_X}$. As mentioned earlier, matrix $\mathbf{U}$ is not generally unitary. The vectors $\mathbf{u}_i$, $i = 1, \ldots, M$, constitute a basis for $\mathbb{C}^M$, albeit a non-orthogonal one, since matrix $\mathbf{R_V}^{-1}\mathbf{R_X}$ is not necessarily Hermitian.

Without loss of generality, it will be assumed that the eigenvalues are arranged in descending order as $\lambda_1 \geq \lambda_2 \geq \ldots \geq \lambda_M \geq 0$ within matrix $\mathbf{\Lambda}$ and the eigenvectors are ordered correspondingly in $\mathbf{U}$.

Eq. (2.11) states that the observed process CPSD matrix is $\mathbf{R_Y} = \mathbf{R_X} + \mathbf{R_V}$, which is full-rank, following $\mathbf{R_V}$. It is easily seen that

$$\mathbf{U}^H\mathbf{R_Y}\mathbf{U} = \mathbf{\Lambda} + \mathbf{I}_M.$$

This means that, if $(\lambda_i, \mathbf{u}_i)$ is an eigenpair of the pencil $(\mathbf{R_X}, \mathbf{R_V})$, then $(\lambda_i + 1, \mathbf{u}_i)$ is an eigenpair of the pencil $(\mathbf{R_Y}, \mathbf{R_V})$, since

$$\mathbf{R_Y}\mathbf{u}_i = \mathbf{R_X}\mathbf{u}_i + \mathbf{R_V}\mathbf{u}_i = \lambda_i\mathbf{R_V}\mathbf{u}_i + \mathbf{R_V}\mathbf{u}_i = (\lambda_i + 1)\mathbf{R_V}\mathbf{u}_i. \tag{4.18}$$

This result can be translated as follows: the GEVD of matrix pencil $(\mathbf{R_X}, \mathbf{R_V})$ can be obtained through the GEVD of matrix pencil $(\mathbf{R_Y}, \mathbf{R_V})$, after adjusting the eigenvalues (by subtracting unity). This is highly important for practical applications since, generally, access to $\mathbf{R_Y}$ or its estimation is easier than that of $\mathbf{R_X}$.

Now, for the left eigenvectors, from Eqs. (4.13) – (4.14) one can see that

$$\mathbf{R}_{\mathbf{X}} = \mathbf{Q}\mathbf{\Lambda}\mathbf{Q}^H = \sum_{i=1}^{M} \lambda_i \mathbf{q}_i \mathbf{q}_i^H \tag{4.19}$$

$$\mathbf{R}_{\mathbf{V}} = \mathbf{Q}\mathbf{Q}^H = \sum_{i=1}^{M} \mathbf{q}_i \mathbf{q}_i^H \tag{4.20}$$

$$\mathbf{R}_{\mathbf{Y}} = \mathbf{Q}\left(\mathbf{\Lambda} + \mathbf{I}\right)\mathbf{Q}^H = \sum_{i=1}^{M} \left(\lambda_i + 1\right) \mathbf{q}_i \mathbf{q}_i^H, \tag{4.21}$$

where $\mathbf{Q} = \mathbf{U}^{-H}$.

From Eq. (4.20), it can be seen that $\mathbf{R}_{\mathbf{V}}^{-1} = \mathbf{U}\mathbf{U}^H$, a fact that reveals that the left and right eigenvectors are related to each other as $\mathbf{U} = \mathbf{U}\mathbf{U}^H\mathbf{Q} = \mathbf{R}_{\mathbf{V}}^{-1}\mathbf{Q}$ or that

$$\mathbf{u}_i = \mathbf{R}_{\mathbf{V}}^{-1}\mathbf{q}_i,$$

for $i = 1, \ldots, M$.

Since $\text{rank}(\mathbf{R}_{\mathbf{X}}) = K$, the ordering of the eigenvalues will in reality be $\lambda_1 \geq \lambda_2 \geq \ldots \geq \lambda_K > \lambda_{K+1} = \ldots = \lambda_M = 0$. In other words, the last $M - K$ generalized eigenvalues are equal to zero. This means that Eq. (4.19) can be adapted and matrix $\mathbf{R}_{\mathbf{X}}$ can be expressed in terms of the first K left eigenvectors only, as

$$\mathbf{R}_{\mathbf{X}} = \sum_{i=1}^{K} \lambda_i \mathbf{q}_i \mathbf{q}_i^H. \tag{4.22}$$

If this equation is looked at in parallel with Eq. (2.12), it is straightforward that the eigenvectors $\mathbf{q}_i$, $i = 1, \ldots, K$ span the same subspace as the target source RTFs $\mathbf{d}_i$, $i = 1, \ldots, K$. Notably, in the case of one target source or $K = 1$, the left eigenvector corresponding to the largest eigenvalue coincides with the steering vector for the source.

To highlight the significance of this result, matrix $\mathbf{U}$ will be partitioned as follows, essentially dividing the eigenvectors into two sets,

$$\mathbf{U} = \begin{bmatrix} \mathbf{U}_1 & \mathbf{U}_2 \end{bmatrix},$$

where

$$\mathbf{U}_1 = \begin{bmatrix} \mathbf{u}_1 & \mathbf{u}_2 & \ldots & \mathbf{u}_K \end{bmatrix}$$

is of size $M \times K$ and

$$\mathbf{U}_2 = \begin{bmatrix} \mathbf{u}_{K+1} & \mathbf{u}_{K+2} & \ldots & \mathbf{u}_M \end{bmatrix}$$

is of size $M \times (M-K)$, with $1 \leq K \leq M$. A similar partitioning of $\mathbf{Q}$ and $\mathbf{\Lambda}$ is implied. Given this, Eq. (4.22) is rewritten as $\mathbf{R}_{\mathbf{X}} = \mathbf{Q}_1\mathbf{\Lambda}_1\mathbf{Q}_1^H$and matrix $\mathbf{R}_{\mathbf{Y}}$ or Eq. (4.21) is partitioned as

$$\mathbf{R}_{\mathbf{Y}} = \begin{bmatrix} \mathbf{Q}_1 & \mathbf{Q}_2 \end{bmatrix} \begin{bmatrix} \mathbf{\Lambda}_1 + \mathbf{I}_K & \mathbf{0} \\ \mathbf{0} & \mathbf{I}_{M-K} \end{bmatrix} \begin{bmatrix} \mathbf{Q}_1^H \\ \mathbf{Q}_2^H \end{bmatrix},$$

where $\mathbf{Q}_1$ is of size $M \times K$ and $\mathbf{Q}_2$ is of size $M \times (M-K)$. Note that $\mathbf{\Lambda}_2 = \mathbf{0}_{(M-K)\times(M-K)}$.

Eq. (4.22) reveals that matrix $\mathbf{Q}_1$ spans the speech subspace. Matrix $\mathbf{U}_2$ spans an orthogonal subspace containing noise only. This follows from Eq. (4.16), since $\mathbf{U}_2^H \mathbf{R}_\mathbf{X} \mathbf{U}_2 = \mathbf{\Lambda}_2 = \mathbf{0}$, which means that for any $\mathbf{u}_j$, $j = K+1, \ldots, M$, that is a column vector of $\mathbf{U}_2$ it is

$$\mathbf{u}_j^H \mathbf{R}_\mathbf{X} \mathbf{u}_j = 0 \Leftrightarrow \mathbb{E}\left[|\mathbf{u}_j^H \mathbf{X}|^2\right] = 0 \Leftrightarrow \mathbf{u}_j^H \mathbf{X} = 0. \tag{4.23}$$

It is, naturally, $\mathbf{Q}_1^H \mathbf{U}_2 = \mathbf{0}_{K\times(M-K)}$, as $\mathbf{Q}^H \mathbf{U} = \mathbf{I}_M$.

Fig. 4.1 shows the relation of these subspaces. The fact that the GEVD provides these matrices justifies why it belongs to the class of signal subspace algorithms. Intuitively, given this decomposition, it is expected that the clean signal can be estimated by nulling the component of the noisy observations residing in the noise subspace [10].

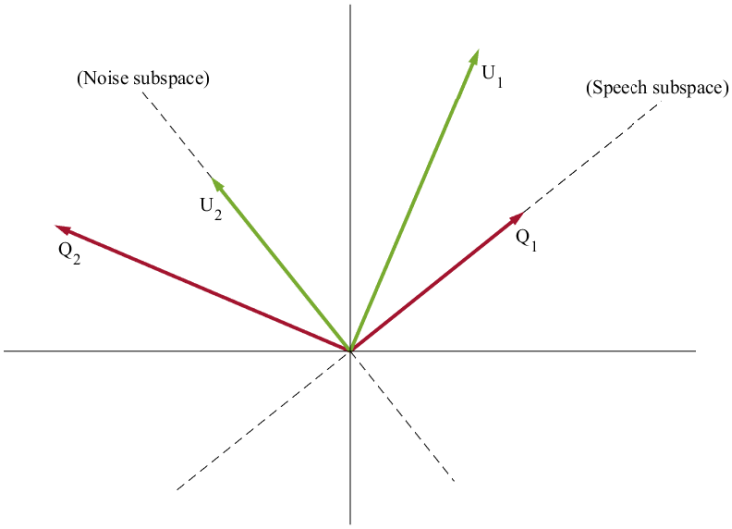


Figure 4.1: Illustration of the speech and noise signal subspaces.

4.4 GEVD-based Optimal Linear Filters

Now that an interpretation for the components delivered by the GEVD is given, in this section, the technique will be exploited in order to formally derive optimal linear beamformers for speech enhancement.

Assume that the received target signal x_1 at reference microphone $m = 1$ is to be estimated from the observation vector $\mathbf{y}$ by means of beamforming, as described in Section 3.1.1. The scalar x_1 is the first element of vector $\mathbf{x}$ and actually stands for the DFT coefficient at the specific frequency bin and time frame. The filtering equations will be developed for the estimation of x_1, but they easily can be adapted for any x_m, $m = 2, \ldots, M$. The output of the beamforming process is

$$\hat{x}_1 = \mathbf{w}^H \mathbf{y},$$

where $\hat{x}_1$ is the estimate of x_1 and $\mathbf{w}$ is the complex-valued filter of length M.

In conventional beamforming techniques, the coefficients w_i, $i = 1, \ldots, M$, are calculated, such as in the delay-and-sum filter and the MVDR filter [40]. However, a different methodology can be utilized: the filter can be expressed in the basis for $\mathbb{C}^M$ formed by the eigenvectors $\mathbf{u}_m$, $m = 1, \ldots, M$, as

$$\mathbf{w} = \mathbf{U}\mathbf{a}, \tag{4.24}$$

where vector $\mathbf{a} = \begin{bmatrix} a_1 & \ldots & a_M \end{bmatrix}^T$ contains the coordinates of $\mathbf{w}$ in the new basis. Instead of w_i, $i = 1, \ldots, M$, the coordinates a_i can be estimated. It is then easy to determine $\mathbf{w}$ from Eq. (4.24).

In order to derive the optimal filter weights, the standard performance criterion of the mean squared-error (MSE) between the beamformer output and the observed target signal will be used, as in works like [37]. The MSE is

$$\begin{aligned} \mathbb{E}\left[|\hat{X}_1 - X_1|^2\right] &= \mathbb{E}\left[|\mathbf{w}^H\mathbf{Y} - X_1|^2\right] \\ &= \mathbb{E}\left[|\mathbf{w}^H\mathbf{X} + \mathbf{w}^H\mathbf{V} - X_1|^2\right] \\ &= \mathbb{E}\left[|\mathbf{w}^H\mathbf{X} - X_1|^2\right] + \mathbb{E}\left[|\mathbf{w}^H\mathbf{V}|^2\right] \end{aligned}$$

since $\mathbb{E}\left[\mathbf{X}\mathbf{V}^H\right] = \mathbf{0}_{M\times M}$, where capital letters denote the respective processes. The term $\mathbb{E}\left[|\mathbf{w}^H\mathbf{X} - X_1|^2\right]$ represents the signal distortion, whereas $\mathbb{E}\left[|\mathbf{w}^H\mathbf{V}|^2\right]$ represents the residual noise variance.

The subsequent derivation is given in [41]. A compromise between signal distortion and noise reduction can be achieved by defining the constrained optimization problem [35, 37, 42]

$$\begin{aligned} \underset{\mathbf{w}}{\text{minimize}} \quad & \mathbb{E}\left[|\mathbf{w}^H\mathbf{X} - X_1|^2\right] \\ \text{subject to} \quad & \mathbb{E}\left[|\mathbf{w}^H\mathbf{V}|^2\right] \leq c, \end{aligned} \tag{4.25}$$

where c is a parameter chosen by the user, with $0 \leq c \leq \sigma_{V_1}^2$, $\sigma_{V_1}^2 = \mathbf{e}_1^H\mathbf{R}_\mathbf{V}\mathbf{e}_1$ is the noise variance at the reference microphone before beamforming and $\mathbf{e}_1 = (1, 0, \ldots, 0)^T$ is the first column of $\mathbf{I}_M$.

Now, since the optimal beamformer is expressed as $\mathbf{w} = \mathbf{U}\mathbf{a}$, the objective function of Eq. (4.25) becomes, using the GEVD,

$$\begin{aligned} \mathbb{E}\left[\left|\mathbf{w}^H\mathbf{X} - X_1\right|^2\right] &= \mathbb{E}\left[\left|\mathbf{w}^H\mathbf{X} - \mathbf{e}_1^H\mathbf{X}\right|^2\right] \\ &= \mathbb{E}\left[\left|\left(\mathbf{U}\mathbf{a} - \mathbf{e}_1\right)^H\mathbf{X}\right|^2\right] \\ &= \mathbf{a}^H\mathbf{U}^H\mathbf{R}_\mathbf{X}\mathbf{U}\mathbf{a} - \mathbf{e}_1^H\mathbf{R}_\mathbf{X}\mathbf{U}\mathbf{a} - \mathbf{a}^H\mathbf{U}^H\mathbf{R}_\mathbf{X}\mathbf{e}_1 + \mathbf{e}_1^H\mathbf{R}_\mathbf{X}\mathbf{e}_1 \\ &= \mathbf{a}^H\mathbf{\Lambda}\mathbf{a} - 2\,\mathrm{Re}\left\{\mathbf{a}^H\mathbf{U}^H\mathbf{R}_\mathbf{X}\mathbf{e}_1\right\} + \sigma_{X_1}^2, \end{aligned}$$

where $\sigma_{X_1}^2 = \mathbf{e}_1^H\mathbf{R}_\mathbf{X}\mathbf{e}_1$ is the target variance at the reference microphone before beamforming.

The residual noise variance becomes

$$\mathbb{E}\left[|\mathbf{w}^H\mathbf{V}|^2\right] = \mathbb{E}\left[\left|(\mathbf{U}\mathbf{a})^H\mathbf{V}\right|^2\right] = \mathbf{a}^H\mathbf{U}^H\mathbf{R}_\mathbf{V}\mathbf{U}\mathbf{a} = \mathbf{a}^H\mathbf{a}.$$

Thus, the problem of Eq. (4.25) is equivalently expressed with respect to $\mathbf{a}$ as

$$\begin{aligned} \underset{\mathbf{a}}{\text{minimize}} \quad & \mathbf{a}^H\mathbf{\Lambda}\mathbf{a} - 2\,\mathrm{Re}\left\{\mathbf{a}^H\mathbf{U}^H\mathbf{R}_\mathbf{X}\mathbf{e}_1\right\} + \sigma_{X_1}^2 \\ \text{subject to} \quad & \mathbf{a}^H\mathbf{a} \leq c. \end{aligned} \tag{4.26}$$

The objective function is a convex function of $\mathbf{a}$. The Lagrangian for this problem is

$$L(\mathbf{a},\mu) = \mathbf{a}^H\mathbf{\Lambda}\mathbf{a} - 2\,\mathrm{Re}\left\{\mathbf{a}^H\mathbf{U}^H\mathbf{R}_\mathbf{X}\mathbf{e}_1\right\} + \sigma_{X_1}^2 + \mu\left(\mathbf{a}^H\mathbf{a} - c\right),$$

with $\mu \geq 0$ a Lagrange multiplier. Let $\mathbf{a}^*$ and μ^* denote the primal and dual optimal, respectively. From the Karush-Kuhn-Tucker (KKT) optimality conditions it must be

$$\mu^*\left(\mathbf{a}^{*H}\mathbf{a}^* - c\right) = 0 \Rightarrow \mu^* = 0 \quad \text{or} \quad \mathbf{a}^{*H}\mathbf{a}^* = c$$

and

$$\nabla_{\bar{\mathbf{a}}} L(\mathbf{a}^*,\mu^*) = 0 \Rightarrow \mathbf{\Lambda}\mathbf{a} - \mathbf{U}^H\mathbf{R}_\mathbf{X}\mathbf{e}_1 + \mu^*\mathbf{a}^* = 0. \tag{4.27}$$

Sticking to the general requirement $\mu \geq 0$, the constraint of Eq. (4.26) is forced to be active at the minimum, meaning there has to hold $\mathbf{a}^{*H}\mathbf{a}^* = c$.

Hence, the minimum of Eq. (4.26) is obtained from Eq. (4.27) as

$$\mathbf{a}^* = \left(\mathbf{\Lambda} + \mu^*\mathbf{I}_M\right)^{-1}\mathbf{U}^H\mathbf{R}_\mathbf{X}\mathbf{e}_1,$$

where μ^* is chosen such that $\mathbf{a}^{*H}\mathbf{a}^* = c$. This means that the optimal beamformers $\mathbf{w}^*$ are

$$\mathbf{w}^* = \mathbf{U}\mathbf{a}^* = \mathbf{U}\left(\mathbf{\Lambda} + \mu^*\mathbf{I}_M\right)^{-1}\mathbf{U}^H\mathbf{R}_\mathbf{X}\mathbf{e}_1$$

Now, from Eqs. (4.19) – (4.20) and since $\mathbf{Q} = \mathbf{U}^{-H}$, it is

$$\mathbf{U}\left(\mathbf{\Lambda} + \mu^*\mathbf{I}_M\right)^{-1}\mathbf{U}^H = \left(\mathbf{U}^{-H}\left(\mathbf{\Lambda} + \mu^*\mathbf{I}_M\right)\mathbf{U}^{-1}\right)^{-1} = \left(\mathbf{R}_\mathbf{X} + \mu^*\mathbf{R}_\mathbf{V}\right)^{-1}.$$

All in all, dropping all * from notation, the optimal beamformers are

$$\mathbf{w} = \left(\mathbf{R}_\mathbf{X} + \mu\mathbf{R}_\mathbf{V}\right)^{-1}\mathbf{R}_\mathbf{X}\mathbf{e}_1 \tag{4.28}$$

$$= \sum_{i=1}^{M}\frac{\mathbf{u}_i\mathbf{u}_i^H}{\lambda_i + \mu}\mathbf{R}_\mathbf{X}\mathbf{e}_1. \tag{4.29}$$

The solution of Eq. (4.28) is referred to as the signal-distortion weighted (SDW) Wiener filter [35, 42] and μ can be seen as a trade-off parameter that controls the signal distortion and noise reduction. Eq. (4.29) expresses the same beamformer in terms of the generalized eigenvectors.

4.5 The GEVD for the Low-Rank Approximation of $\mathbf{R_X}$

The analysis until here has assumed that there is direct access to the actual CPSD matrices of the processes. This is hardly ever the case, though, and these matrices are in practice estimated using the data available, as explained in Section 2.4.1.

In most cases, the target CPSD matrix is estimated as $\hat{\mathbf{R}}_\mathbf{X} = \hat{\mathbf{R}}_\mathbf{Y} - \hat{\mathbf{R}}_\mathbf{V}$, as Eq. (2.13) states. When there are K target speech sources and the MTF approximation is valid, it will be $\text{rank}\,(\mathbf{R_X}) = K$. However, in a practical implementation it will be $\text{rank}(\hat{\mathbf{R}}_\mathbf{X}) > K$ due to disturbances such as longer reverberation, microphone self-noise and estimation inaccuracies. Additionally, using Eq. (2.13) does not guarantee positive semi-definiteness for $\hat{\mathbf{R}}_\mathbf{X}$. This is especially true in high-noise scenarios and has been observed to lead to unpredictable noise reduction performance [43]. That is why in certain cases it is desired to replace $\mathbf{R_X}$ by a low-rank approximation of it and not by the difference of other estimates.

Based on Eq. (4.19), which states that $\mathbf{R_X} = \sum_{i=1}^{M} \lambda_i \mathbf{q}_i \mathbf{q}_i^H$, a rank-$R$ approximation of $\mathbf{R_X}$ can be computed in practice as such: firstly, the estimates $\hat{\mathbf{R}}_\mathbf{V}$ and $\hat{\mathbf{R}}_\mathbf{Y}$ are calculated routinely. Then, the GEVD of matrix pencil $(\hat{\mathbf{R}}_\mathbf{Y}, \hat{\mathbf{R}}_\mathbf{V})$ is performed and the generalized eigenpairs $(\lambda_i + 1, \mathbf{u}_i)$, $i = 1, \ldots, M$ are found (see Eq. (4.18)).

An R-rank approximation of $\mathbf{R_X}$ can be then obtained by selecting only the first R adjusted eigenvalues, as in

$$\hat{\mathbf{R}}_\mathbf{X} = \sum_{r=1}^{R} \lambda_r \mathbf{q}_r \mathbf{q}_r^H. \tag{4.30}$$

In case $R = K$, a good estimate of the real CPSD matrix is expected. The choice of R can be based on prior information about the number of sources or by inspecting the singular values delivered by the SVD of $\hat{\mathbf{R}}_\mathbf{Y} - \hat{\mathbf{R}}_\mathbf{V}$.

This technique implies that the last eigenvalues are set to zero, especially eigenvalues that are possibly negative, a tactic also described in [44] and considered in [43, 45].

4.6 Optimal Variable Span Linear Filters

In this section, a convenient framework that manages to group together all optimal linear beamformers will be described, using the findings of Section 4.4. This framework will later be exploited in Chapter 5 to construct a mathematical proof.

When the optimal noise reduction filters $\mathbf{w} = \mathbf{U}\mathbf{a}$ are designed using the actual CPSD matrices with at most $\text{rank}(\mathbf{R_X}) = K$ constraints, this will lead to filter coefficients $a_i = 0$ for $i = K+1, \ldots, M$, since there is no speech in the direction of $\mathbf{U}_2$, as Eq. (4.23) and Fig. 4.1 indicate.

With this remark as motivation, it is argued in [37] that a more flexible linear filter can be defined by forcing any desired number of coefficients to zero and choosing

$$\mathbf{w}(P) = \mathbf{U}_P \mathbf{a}_P,$$

where $\mathbf{U}_P = \begin{bmatrix}\mathbf{u}_1 & \mathbf{u}_2 & \dots & \mathbf{u}_P\end{bmatrix}$ and $\mathbf{a}_P = \begin{bmatrix}a_1 & \dots & a_P\end{bmatrix}^T$.

These beamformers only use the first P eigenvectors contained in $\mathbf{U}$ and implicitly force the last $M-P$ elements of $\mathbf{a}$ to 0, although this generally holds only when $P = K$. The resulting filter $\mathbf{w}(P)$ is called a variable span (VS) linear filter [37] of length M and it is $\mathbf{w}(P) \in \text{Span}\{\mathbf{u}_1, \mathbf{u}_2, \dots, \mathbf{u}_P\}$. Hence the name of the filter, as adjusting parameter P causes the filter to lie in a different vector span.

Given the general form of optimal filters in Eqs. (4.28) – (4.29), the general form of an optimal VS filter is written here as follows [37]

$$\mathbf{w}(P) = \mathbf{U}_P\mathbf{a}_P = \mathbf{U}_P(\mathbf{\Lambda}_P + \mu\mathbf{I}_P)^{-1}\mathbf{U}_P^H\mathbf{R}_\mathbf{X}\mathbf{e}_1 \tag{4.31}$$

$$= \sum_{i=1}^{P} \frac{\mathbf{u}_i\mathbf{u}_i^H}{\lambda_i + \mu}\mathbf{R}_\mathbf{X}\mathbf{e}_1, \tag{4.32}$$

where μ is a tradeoff parameter that controls the signal distortion and noise reduction and $\mathbf{e}_1$ is the first column of $\mathbf{I}_M$.

The estimate of the target signal at the output of the beamformer is

$$\begin{aligned}\hat{x}_1 &= \mathbf{w}^H(P)\mathbf{y} \\ &= \mathbf{e}_1^T\mathbf{R}_\mathbf{X}\mathbf{U}_P(\mathbf{\Lambda}_P + \mu\mathbf{I}_P)^{-1}\mathbf{U}_P^H\mathbf{y}.\end{aligned}$$

By manipulating parameters P and μ, it is possible to build a wide range of filters with controlled characteristics. The ensuing beamformers are broadly categorized as follows and a summary of them is is given in Table 4.1:

- Setting $\mu = 0$ and $P \leq K$ leads to VS minimum distortion filters. The MVDR filter belongs here and is obtained for $P = K$.
- Setting $\mu = 1$ leads to VS Wiener filters. The classical multi-channel Wiener filter belongs here and is obtained for $P = M$.
- Setting $0 \leq \mu \leq 1$ leads to VS trade-off filters. The classical trade-off filter belongs here and is obtained for $P = M$.

4.6.1 Optimal Vs Linear Filters with Low-Rank Approximation of $\mathbf{R}_\mathbf{X}$

Combining the findings of Section 4.5 and Eqs. (4.31) – (4.32) it is possible to derive directly optimal VS beamformers that use an R-rank approximation of $\mathbf{R}_\mathbf{X}$ as

$$\begin{aligned}\mathbf{w}_{appr}(P,R) &= \mathbf{U}_P(\mathbf{\Lambda}_P + \mu\mathbf{I}_P)^{-1}\mathbf{U}_P^H\hat{\mathbf{R}}_\mathbf{X}\mathbf{e}_1 \\ &= \sum_{i=1}^{P}\frac{\mathbf{u}_i\mathbf{u}_i^H}{\lambda_p + \mu}\left(\sum_{r=1}^{R}\lambda_r\mathbf{q}_r\mathbf{q}_r^H\right)\mathbf{e}_1 \\ &= \sum_{i=1}^{P}\frac{\mathbf{u}_i\mathbf{u}_i^H}{\lambda_p + \mu}\left(\sum_{r=1}^{R}\lambda_r q_r^*(1)\mathbf{q}_r\right) \\ &= \sum_{i=1}^{\min(P,R)}\frac{\lambda_i}{\lambda_i + \mu}q_i^*(1)\mathbf{u}_i,\end{aligned} \tag{4.33}$$

Table 4.1: Optimal Variable Span Linear Filters.

Name			Expression	
MVDR	$\mathbf{w}_{\mathrm{MVDR}}$	$=$	$\sum_{i=1}^{K} \frac{\mathbf{u}_i \mathbf{u}_i^H}{\lambda_i} \mathbf{R}_{\mathbf{X}} \mathbf{e}_1$	
Wiener	$\mathbf{w}_{\mathrm{W}}$	$=$	$\sum_{i=1}^{M} \frac{\mathbf{u}_i \mathbf{u}_i^H}{\lambda_i + 1} \mathbf{R}_{\mathbf{X}} \mathbf{e}_1$	
Tradeoff	$\mathbf{w}_{\mathrm{T},\mu}$	$=$	$\sum_{i=1}^{M} \frac{\mathbf{u}_i \mathbf{u}_i^H}{\lambda_i + \mu} \mathbf{R}_{\mathbf{X}} \mathbf{e}_1$	
VS Minimum Distortion	$\mathbf{w}_{\mathrm{MD}}(P)$	$=$	$\sum_{i=1}^{P} \frac{\mathbf{u}_i \mathbf{u}_i^H}{\lambda_i} \mathbf{R}_{\mathbf{X}} \mathbf{e}_1,$	$P \leq K$
VS Wiener	$\mathbf{w}_{\mathrm{W}}(P)$	$=$	$\sum_{i=1}^{P} \frac{\mathbf{u}_i \mathbf{u}_i^H}{\lambda_i + 1} \mathbf{R}_{\mathbf{X}} \mathbf{e}_1,$	$P \leq M$
VS Tradeoff	$\mathbf{w}_{\mathrm{T},\mu}(P)$	$=$	$\sum_{i=1}^{P} \frac{\mathbf{u}_i \mathbf{u}_i^H}{\lambda_i + \mu} \mathbf{R}_{\mathbf{X}} \mathbf{e}_1,$	$\mu \geq 0$

due to Eq. (4.15). Note that $\mathbf{q}_r^H \mathbf{e}_1 = q_r^*(1)$, which is the first element of vector $\mathbf{q}_r^*$.

Eq. (4.33) reveals that, if $\mathbf{R}_{\mathbf{X}}$ is approximated by a low-rank version $\hat{\mathbf{R}}_{\mathbf{X}}$ using the GEVD the optimal VS filters that result are straightforward linear combinations of the generalized eigenvectors.

5 GEVD for Clock Offset Model

This chapter epitomizes this thesis; the system model for WASNs facing sensor clock offsets is developed and the signal subspace methodology of Chapter 4 is applied to this model. The end goal is to manifest the clock-offset invariant nature of GEVD-based blind beamforming techniques.

5.1 System Model Including Clock Offsets

In order to build the mathematical model for systems who face clock offsets, the offset of each microphone with respect to the reference microphone $m = 1$ is denoted τ_m, $m = 1, 2, \ldots, M$, and $\tau_m \in \mathbb{R}$ and it considered unchanging during the function of the network. Note that $\tau_1 = 0$.

A positive clock offset implies that data entries registered with a specific time tick were in reality collected at an earlier time than the entries of the reference microphone bearing the same time tick. On the other hand, a negative clock offset means that they were actually collected at a later time than them. This suggests that the new system model with non-synchronized clocks can be built as follows: a positive offset is equivalent to translating the specific microphone signal of the synchronized clock model later in time and a negative one is equivalent to shifting it earlier in time.

It should be stressed that a clock offset, as described in this thesis, characterized a specific microphone. This means that all signals recorded by the sensor are affected in the same way, regardless of their origin.

Regarding notation, all quantities bearing a tilde will refer to the system that includes clock offsets.

Thus, the continuous-time signal $\tilde{y}_m(t)$, $m = 1, 2, ..., M$, at the output of the m-th microphone results from Eq. (2.8) after applying a translation in time as follows

$$\begin{aligned}\tilde{y}_m(t) = y_m(t+\tau_m) &= \sum_{n=1}^{K} h_{n,m}(t+\tau_m) * s_n(t) + \sum_{n=K+1}^{N_s} h_{n,m}(t+\tau_m) * s_n(t) + z_m(t+\tau_m) \\ &= x_m(t+\tau_m) + v_m(t+\tau_m) \qquad (5.1) \\ &= \tilde{x}_m(t) + \tilde{v}_m(t). \qquad (5.2)\end{aligned}$$

where $\tilde{x}_m(t) = \sum_{n=1}^{K} h_{n,m}(t+\tau_m) * s_n(t)$ is the observed target signal at microphone m, and $\tilde{v}_m(t) = \sum_{n=K+1}^{N_s} h_{n,m}(t+\tau_m) * s_n(t) + n_m(t+\tau_m)$ is the total noise signal. All signals are, again, considered realizations of zero-mean WSS processes.

Note that for the STFT representation of Eq. (5.1) to be obtained, the MTF approximation should still hold. This means that the support of the RIRs including the clock offsets has to be small compared to the window length.

Before proceeding, it is important to remind the reader of the following property of the Fourier transform regarding time shifts

- If signal $g(t)$ is sampled and the DFT $G(f)$ is obtained, then the same process for $g(t+t_0)$ will give $G(f)\mathrm{e}^{j2\pi f t_0}$, where f is the frequency variable. This means that only the phase of the DFT will be affected.

Using this property, Eq. (2.8) is rewritten into the STFT domain using a window of length N_{DFT}, as in

$$\begin{aligned}\tilde{Y}_m(k,l) &= \sum_{n=1}^{K} H_{m,n}(l)S_n(k,l)\mathrm{e}^{j2\pi f_k\tau_m} + \sum_{n=K+1}^{N_s} H_{m,n}(l)S_n(k,l)\mathrm{e}^{j2\pi f_k\tau_m} + Z_m(k,l)\mathrm{e}^{j2\pi f_k\tau_m} \\ &= X_m(k,l)\mathrm{e}^{j2\pi f_k\tau_m} + V_m(k,l)\mathrm{e}^{j2\pi f_k\tau_m}, \end{aligned} \tag{5.3}$$

where $f_k = k/N$ and N is number of frequency components. Stacking the M phase difference components in a vector and dropping k will give the clock phase vector

$$\boldsymbol{\tau}(f) = \begin{bmatrix}\mathrm{e}^{j2\pi f\tau_1} & \mathrm{e}^{j2\pi f\tau_2} & \dots & \mathrm{e}^{j2\pi f\tau_M}\end{bmatrix}. \tag{5.4}$$

Dropping f from the notation, the vector representation of the observed signal in the frequency domain, using Eqs. (2.10) and (5.4), becomes

$$\begin{aligned}\tilde{\mathbf{y}} &= \mathbf{x}\circ\boldsymbol{\tau} + \mathbf{v}\circ\boldsymbol{\tau} \\ &= \sum_{n=1}^{K}(\mathbf{h}_n\circ\boldsymbol{\tau})\,S_n + \mathbf{v}\circ\boldsymbol{\tau} \\ &= \sum_{n=1}^{N_s}(\mathbf{d}_n\circ\boldsymbol{\tau})\,Z_{1,n} + \mathbf{v}\circ\boldsymbol{\tau} \\ &= \sum_{n=1}^{N_s}\tilde{\mathbf{d}}_n Z_{1,n} + \mathbf{v}\circ\boldsymbol{\tau}, \end{aligned} \tag{5.5}$$

where $\circ$ represents the Hadamard product operator (element-wise multiplication).

The above demonstrates that the ATF vector when clock offsets are present is

$$\tilde{\mathbf{h}}_n = \mathbf{h}_n\circ\boldsymbol{\tau} \tag{5.6}$$

and the RTF vector then is

$$\begin{aligned}\tilde{\mathbf{d}}_n &= \frac{1}{H_{1,n}}\mathbf{h}_n\circ\boldsymbol{\tau} \\ &= \mathbf{d}_n\circ\boldsymbol{\tau}. \end{aligned} \tag{5.7}$$

In other words, when sensors exhibit clock offsets, their ATFs (and RTFs) undergo a transformation, given away by Eqs. (5.6) – (5.7) : they are perceived as having the same amplitude but an adjusted phase component. This is as if the sensor was in the far-field of a source (where attenuation is constant for all locations) and it was relocated to a new position.

Now, a new matrix $\mathbf{T}$ that contains the phase difference elements in its diagonal will be defined as

$$\begin{aligned}\mathbf{T} &= \begin{bmatrix} \mathrm{e}^{j2\pi f\tau_1} & 0 & \dots & 0 \\ 0 & \mathrm{e}^{j2\pi f\tau_2} & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & \mathrm{e}^{j2\pi f\tau_M} \end{bmatrix} \\ &= \begin{bmatrix} 1 & 0 & \dots & 0 \\ 0 & \mathrm{e}^{j2\pi f\tau_2} & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & \mathrm{e}^{j2\pi f\tau_M} \end{bmatrix}\end{aligned}$$

This matrix carries the significant property of being unitary, meaning

$$\mathbf{T}\mathbf{T}^H = \mathbf{T}^H\mathbf{T} = \mathbf{I}. \tag{5.8}$$

Using $\mathbf{T}$, another way to express Eq. (5.5) is

$$\tilde{\mathbf{y}} = \mathbf{T}\mathbf{x} + \mathbf{T}\mathbf{v}. \tag{5.9}$$

Eq. (5.9) clearly shows that, when clock offset exist in the system, the new received process, the new received target process and the new received noise process, compared to the synchronized system, are, respectively,

$$\begin{aligned}\tilde{\mathbf{Y}} &= \mathbf{T}\mathbf{Y} \\ \tilde{\mathbf{X}} &= \mathbf{T}\mathbf{X} \\ \tilde{\mathbf{V}} &= \mathbf{T}\mathbf{V}\end{aligned}$$

Due to Eq. (5.8), the CPSD matrices of the new processes are

$$\tilde{\mathbf{R}}_{\mathbf{Y}} = \mathbb{E}[\tilde{\mathbf{Y}}\tilde{\mathbf{Y}}^H] = \mathbb{E}[(\mathbf{T}\mathbf{Y})(\mathbf{T}\mathbf{Y})^H] = \mathbf{T}\mathbf{R}_{\mathbf{Y}}\mathbf{T}^H \tag{5.10}$$

$$\tilde{\mathbf{R}}_{\mathbf{X}} = \mathbb{E}[\tilde{\mathbf{X}}\tilde{\mathbf{X}}^H] = \mathbf{T}\mathbf{R}_{\mathbf{X}}\mathbf{T}^H \tag{5.11}$$

$$\tilde{\mathbf{R}}_{\mathbf{V}} = \mathbb{E}[\tilde{\mathbf{V}}\tilde{\mathbf{V}}^H] = \mathbf{T}\mathbf{R}_{\mathbf{V}}\mathbf{T}^H. \tag{5.12}$$

As apparent from Eqs. (5.10) – (5.12), all CPSD matrices corresponding to the new processes are unitarily similar to the ones of the synchronized system, with $\mathbf{T}$ as the base change matrix. That means the respective matrices share the same rank and eigenvalues, including their multiplicity. The respective eigenspaces are connected through the base change matrix $\mathbf{T}$.

5.2 The GEVD for the Clock Offset Model

In this section, the GEVD for the new system matrices of the received signal model $\tilde{\mathbf{y}} = \mathbf{T}\mathbf{x} + \mathbf{T}\mathbf{v}$ of Eq. (5.9) will be formed. Again, it is assumed that all target speech signals are incorporated in $\mathbf{x}$, whereas the effect of interfering signals is absorbed within $\mathbf{v}$. The aim is to understand the effect the phase difference of the RTFs has on the beamforming process.

As explained in Section 4.3, the GEVD of the matrix pencil $(\mathbf{R}_\mathbf{X}, \mathbf{R}_\mathbf{V})$ is equivalent to the EVD of the matrix product $\mathbf{R}_\mathbf{V}^{-1}\mathbf{R}_\mathbf{X}$. As a reminder, this EVD gives $\mathbf{R}_\mathbf{V}^{-1}\mathbf{R}_\mathbf{X} = \mathbf{U}\mathbf{\Lambda}\mathbf{U}^{-1} = \mathbf{Q}^{-H}\mathbf{\Lambda}\mathbf{Q}^H$, where $\mathbf{U}$ contains the right eigenvectors, $\mathbf{Q}$ the left ones and $\mathbf{\Lambda}$ the eigenvalues in its diagonal.

This matrix product for sensors exhibiting offsets becomes, using Eqs. (5.10) – (5.12),

$$\begin{aligned}\tilde{\mathbf{R}}_\mathbf{V}^{-1}\tilde{\mathbf{R}}_\mathbf{X} &= \left(\mathbf{T}\mathbf{R}_\mathbf{V}\mathbf{T}^H\right)^{-1}\left(\mathbf{T}\mathbf{R}_\mathbf{X}\mathbf{T}^H\right) \\ &= \mathbf{T}\mathbf{R}_\mathbf{V}^{-1}\mathbf{R}_\mathbf{X}\mathbf{T}^H. \end{aligned} \tag{5.13}$$

where $\mathbf{T}^{-1} = \mathbf{T}^H$ was used.

As apparent from Eq. (5.13), matrix $\tilde{\mathbf{R}}_\mathbf{V}^{-1}\tilde{\mathbf{R}}_\mathbf{X}$ is also unitarily similar to the one of the synchronized system, with $\mathbf{T}$ as the base change matrix.

Now, Eq. (5.13) will be rewritten using the GEVD of the initial matrix pencil

$$\begin{aligned}\tilde{\mathbf{R}}_\mathbf{V}^{-1}\tilde{\mathbf{R}}_\mathbf{X} &= \mathbf{T}\mathbf{R}_\mathbf{V}^{-1}\mathbf{R}_\mathbf{X}\mathbf{T}^H \\ &= \mathbf{T}\left(\mathbf{U}\mathbf{\Lambda}\mathbf{U}^{-1}\right)\mathbf{T}^H \\ &= \left(\mathbf{T}\mathbf{U}\right)\mathbf{\Lambda}\left(\mathbf{T}\mathbf{U}\right)^{-1}. \end{aligned} \tag{5.14}$$

Eq. (5.14) shows the EVD of the new matrix product, with matrix $\tilde{\mathbf{U}} = \mathbf{T}\mathbf{U}$ containing the new right eigenvectors. The eigenvalues in $\mathbf{\Lambda}$ remain the same as when the clocks are perfectly synchronized.

As for the left eigenvectors, the new matrix is $\tilde{\mathbf{Q}} = \mathbf{T}\mathbf{Q}$, since it is

$$\begin{aligned}\tilde{\mathbf{R}}_\mathbf{V}^{-1}\tilde{\mathbf{R}}_\mathbf{X} &= \mathbf{T}\mathbf{R}_\mathbf{V}^{-1}\mathbf{R}_\mathbf{X}\mathbf{T}^H \\ &= \mathbf{T}\left(\mathbf{Q}^{-H}\mathbf{\Lambda}\mathbf{Q}^H\right)\mathbf{T}^H \\ &= \left(\mathbf{T}\mathbf{Q}\right)^{-H}\mathbf{\Lambda}\left(\mathbf{T}\mathbf{Q}\right)^H. \end{aligned} \tag{5.15}$$

Eqs. (5.14) – (5.15) provide the highly important result that the GEVD for the clock offset system model will give

$$\begin{aligned}\tilde{\mathbf{U}}^H\tilde{\mathbf{R}}_\mathbf{X}\tilde{\mathbf{U}} &= \mathbf{\Lambda} \\ \tilde{\mathbf{U}}^H\tilde{\mathbf{R}}_\mathbf{V}\tilde{\mathbf{U}} &= \mathbf{I}_M \\ \tilde{\mathbf{R}}_\mathbf{X} &= \tilde{\mathbf{Q}}\mathbf{\Lambda}\tilde{\mathbf{Q}}^H \\ \tilde{\mathbf{R}}_\mathbf{V} &= \tilde{\mathbf{Q}}\tilde{\mathbf{Q}}^H \\ \tilde{\mathbf{R}}_\mathbf{Y} &= \tilde{\mathbf{Q}}\left(\mathbf{\Lambda} + \mathbf{I}\right)\tilde{\mathbf{Q}}^H. \end{aligned}$$

It should be noted here that, as mentioned in Section 4.2, the eigenvectors are generally not unique. This means matrix $\tilde{\mathbf{U}} = \mathbf{TU}$ will not necessarily be the result of the GEVD algorithm, if it ran on new data. If $\mathbf{D}$ is any diagonal unitary matrix of size $M \times M$, meaning $\mathbf{D} = \text{diag}(\text{e}^{jd_1}, \text{e}^{jd_2}, \cdots, \text{e}^{jd_M})$ with $\mathbf{D}^H\mathbf{D} = \mathbf{D}\mathbf{D}^H = \mathbf{I}_M$, it is clear that

$$(\mathbf{TUD})^H\mathbf{R}_\mathbf{X}(\mathbf{TUD}) = \mathbf{D}^H\mathbf{\Lambda}\mathbf{D} = \mathbf{\Lambda}, \quad \text{and} \quad (\mathbf{TUD})^H\mathbf{R}_\mathbf{V}(\mathbf{TUD}) = \mathbf{D}^H\mathbf{I}_M\mathbf{D} = \mathbf{I}_M.$$

Therefore, any matrices of the form $\tilde{\mathbf{U}} = \mathbf{TUD}$ and $\tilde{\mathbf{Q}} = \mathbf{TQD}$ will provide a generalized eigenvalue decomposition for the pencil $(\tilde{\mathbf{R}}_\mathbf{X}, \tilde{\mathbf{R}}_\mathbf{V})$.

5.3 Optimal Linear Filters for the Clock Offset Model

Now, the most general form of optimal linear filters $\mathbf{w}(P) = \mathbf{U}_P(\mathbf{\Lambda}_P + \mu\mathbf{I}_P)^{-1}\mathbf{U}_P^H\mathbf{R}_\mathbf{X}\mathbf{e}_1$ of Eq. (4.31), with P the number of eigenvectors included, will be worked upon for the model including sensor clock offsets. For this, matrix $\tilde{\mathbf{R}}_\mathbf{X} = \mathbf{T}\mathbf{R}_\mathbf{X}\mathbf{T}^H$ will be used together with the first P generalized eigenvectors contained in matrix $\tilde{\mathbf{U}}_P = \mathbf{TU}_P$. The optimal VS filters, in this case, are given by

$$\begin{aligned}
\tilde{\mathbf{w}}(P) &= \tilde{\mathbf{U}}_P(\mathbf{\Lambda}_P + \mu\mathbf{I}_P)^{-1}\tilde{\mathbf{U}}_P^H\tilde{\mathbf{R}}_\mathbf{X}\mathbf{e}_1 \\
&= (\mathbf{TU}_P)(\mathbf{\Lambda}_P + \mu\mathbf{I}_P)^{-1}(\mathbf{TU}_P)^H(\mathbf{T}\mathbf{R}_\mathbf{X}\mathbf{T}^H)\mathbf{e}_1 \\
&= \mathbf{TU}_P(\mathbf{\Lambda}_P + \mu\mathbf{I}_P)^{-1}\mathbf{U}_P^H\mathbf{R}_\mathbf{X}\mathbf{e}_1 \\
&= \mathbf{T}\mathbf{w}(P),
\end{aligned}$$

where $\mathbf{w}(P)$ is the beamformer given in Eq. (4.31) for perfectly synchronized clocks, since $\mathbf{T}^H\mathbf{T} = \mathbf{I}_M$ and $\mathbf{T}^H\mathbf{e}_1 = \mathbf{e}_1$. This result is easily proved to hold for any matrix $\tilde{\mathbf{U}} = \mathbf{TUD}$ of Section 5.2.

The output of the beamforming operation in this case is

$$\begin{aligned}
\hat{x}_1 &= \tilde{\mathbf{w}}^H(P)\tilde{\mathbf{y}} \\
&= (\mathbf{T}\mathbf{w}(P))^H(\mathbf{T}\mathbf{y}) \\
&= \mathbf{w}^H(P)\mathbf{y}.
\end{aligned} \tag{5.16}$$

Eq. (5.16) reveals that GEVD-based beamformers are invariant to clock offsets and produce the same target estimate as if the clocks were perfectly synchronized. This means that the beamformer itself deals with the offsets and there is no need to explicitly estimate and compensate for them. As typical beamformers, such as the MVDR and the Wiener filters can be expressed in terms of the generalized eigenvectors, as desibred in Section 4.6, it is suggested that in general blind beamforming techniques are invariant to sensor clock offsets.

Going back to Fig. 3.2 of Section 3.3, if the beamformer is formed by the GEVD based on prior data and then applied to offset-affected data, its output will be $\mathbf{w}^H(P)\tilde{\mathbf{y}} = \mathbf{w}^H(P)\mathbf{T}\mathbf{y} \neq \hat{x}_1$. That is why the location-based RTFs do not give a satisfying noise reduction performance when sensor offsets are present.

5.3.1 Optimal Filters with Low-Rank Approximation of $\mathbf{R_X}$

The decompositions $\tilde{\mathbf{U}} = \mathbf{TU}$ and $\tilde{\mathbf{Q}} = \mathbf{TQ}$ reveal that new generalized eigenvectors are given by $\tilde{\mathbf{u}}_p = \mathbf{T}\mathbf{u}_p$ and $\tilde{\mathbf{q}}_p = \mathbf{T}\mathbf{q}_p$, where $\mathbf{u}_P$ and $\mathbf{q}_p$ are the eigenvectors for perfectly synchronized clocks.

When clock offsets are present in the network, the optimal filters with an R-rank approximation of $\tilde{\mathbf{R}}_{\mathbf{X}}$ result from Eq. (4.33) as

$$\begin{aligned}
\tilde{\mathbf{w}}_{approx}(P,R) &= \sum_{p=1}^{\min(P,R)} \frac{\lambda_p}{\lambda_p+\mu}\tilde{q}_p^*(1)\tilde{\mathbf{u}}_p \\
&= \sum_{p=1}^{\min(P,R)} \frac{\lambda_p}{\lambda_p+\mu}\left(T_{1,1}^* q_p^*(1)\right)(\mathbf{T}\mathbf{u}_p) \\
&= \mathbf{T}\sum_{p=1}^{\min(P,R)} \frac{\lambda_p}{\lambda_p+\mu} q_p^*(1)\mathbf{u}_p \\
&= \mathbf{T}\mathbf{w}_{approx}(P,R).
\end{aligned}$$

since $T_{1,1} = 1$.

The output of the beamforming operation in this case is

$$\begin{aligned}
\hat{x}_1 &= \tilde{\mathbf{w}}_{approx}^H(P,R)\tilde{\mathbf{y}} \\
&= \left(\mathbf{T}\mathbf{w}_{approx}(P,R)\right)^H(\mathbf{T}\mathbf{y}) \\
&= \mathbf{w}_{approx}^H(P,R)\mathbf{y}. \qquad (5.17)
\end{aligned}$$

Eq. (5.16) shows that GEVD-based beamformers with a low-rank approximation of $\tilde{\mathbf{R}}_{\mathbf{X}}$ once again take care of the offsets, sparing the need to estimate them explicitly.

5.4 Practical implementation

As affirmed in Eqs. (5.10) – (5.12), the CPSD matrices corresponding to processes including offsets are unitarily similar to the ones of the synchronized system, as for example in $\tilde{\mathbf{R}}_{\mathbf{Y}} = \mathbf{T}\mathbf{R}_{\mathbf{Y}}\mathbf{T}^H$.

In an practical scenario, the CPSD matrices have to be estimated, as explained in Section 2.4.1. In this implementation, the sample covariance matrices of the same section are utilized. The estimate for the received process CPSD matrix in a synchronized system is repeated here

$$\hat{\mathbf{R}}_{\mathbf{Y}} = \frac{1}{N_Y}\sum_{n=1}^{N_Y}\mathbf{y}(n)\mathbf{y}^H(n),$$

where $\mathbf{y}(n)$ is the observed signal in time frame n and N_Y is the number of frames used.

For the mathematical analysis in Section 5.3 to hold precisely and the GEVD-based beamformers to be clock-offset invariant in a practical implementation, the necessary condition is for offset-affected estimated CPSD matrices to be unitarily similar to the estimated ones of the synchronized system. If the sample covariance matrix for the offset-affected received process is denoted $\mathcal{R}_{\mathbf{Y}}$, this condition translates to

$$\begin{aligned}\mathcal{R}_{\mathbf{Y}} &= \mathbf{T}\left(\frac{1}{N_Y}\sum_{n=1}^{N_Y}\mathbf{y}(n)\mathbf{y}^H(n)\right)\mathbf{T}^H \\ &= \frac{1}{N_Y}\sum_{n=1}^{N}\left(\mathbf{T}\mathbf{y}(n)\right)\left(\mathbf{T}\mathbf{y}(n)\right)^H.\end{aligned}$$

This extends to the other system processes, as well.

This implies that, when clock offsets are present, the microphone measurement vector at time instant n should be

$$\tilde{\mathbf{y}}(n) = \mathbf{T}\mathbf{y}(n). \tag{5.18}$$

Is this claim, though, true?

Suppose that under absolute clock synchronization the set of time frames during the processing of the signal is $\{F_i\}$, $i = 1, \ldots, N_t$, where N_t is the total number of frames.

In a WASN with clock offsets, the received signals at the microphones are perceived as having been shifted in time. The new set of frames during processing is denoted $\{F_i'\}$, $i = 1, \ldots, N_t'$, where N_t' is the new total number of frames, possibly different than before.

However, the time signal in frame F_n' with frequency representation $\tilde{\mathbf{y}}(n)$ will not be a shifted version of the time signal in the respective frame F_n with frequency representation $\mathbf{y}(n)$. Due to the windowing operation of the STFT, some samples that were found in frame F_{n-1} or frame F_{n+1} in the initial signal will be found under frame F_n' after the translation in time is applied, depending on the direction of the translation.

Therefore, Eq. (5.18) does not hold exactly and the filters cannot achieve the exact same noise reduction in the two cases. Nevertheless, the differences will be small and expected to be mitigated if longer windows are used for the analysis.

Results in Simulated Environment

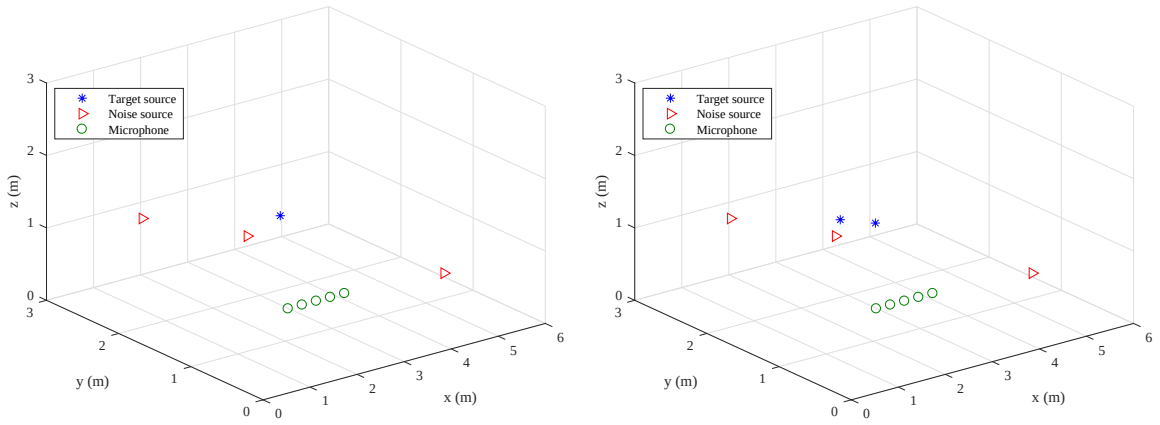
6

In Chapter 5, it was proved how blind beamforming techniques are invariant to clock offsets, when there is direct access to the CPSD matrices of the processes. In this chapter, the results of tests conducted in a simulated environment are provided, to support the claims of the previous chapter and to understand the effect the frame-by-frame processing has on the results, as explained in Section 5.4.

6.1 Simulations Set-up

For the simulations, a room with dimensions $6m \times 3m \times 3m$ is considered. The experiments are conducted for one or two target sources and a fixed number of interfering sources, equal to 3. The number of microphones is fixed to $M = 5$ and they are placed in a line, at a distance of $30cm$ to each other. This room setup is illustrated in Fig. 6.1a for one target source and Fig. 6.1b for two target sources. The sensor self-noise is created at $30dB$ level lower than the microphone measured signal.

For the calculation of the RIRs, the RIR Generator toolbox [46] of Emanuël A.P. Habets for Matlab® is used, which adopts the geometrical room acoustics assumptions of Section 2.2 and, specifically, uses the image source method [47]. The reflection coefficients are considered frequency-independent for all walls. The enclosure is considered empty, in that no object possibly inside it is modeled. The microphones are omnidirectional, with orientation at 0 degrees. The sampling frequency of all sensors is $F_s = 16kHz$.



(a) Set-up with one target source.

(b) Set-up with two target sources.

Figure 6.1: Room set-up for simulations.

For the signal statistics, the sample covariance matrices, as given in Section 2.4.1 were used. For the noise statistics, it is assumed that there is a long period at the beginning (of $5s$) where the target sources are inactive. To identify that, a VAD in the system is assumed. The time samples of Section 2.4.1 were obtained by applying the STFT with a rectangular window of $32ms$ duration and 50% overlap. For the beamforming, the STFT was applied using a square root Hann analysis window and a square root Hann synthesis window (see Appendix A.3). An overview of the beamformers is given in Table 4.1. To permit the MTF approximation, the RIR are considered to have a duration of $12ms$, which is shorter that half the length of the window.

Clock offsets were introduced in the system by shifting the synchronized signal in time, as explained in Section 5.1, for microphones $m = 2, \ldots, 5$. The clock offset values for all results are given in number of samples. The beamformers considered are two, namely the Wiener filter and the trade-off filter with with $\mu = 0.2$ and an R-rank approximation of the $\mathbf{R_X}$ for $R = 3$.

To measure the beamformer performance, the broadband output SNR is plotted over the broadband input SNR, as calculated at the reference microphone $m = 1$. The input SNR is defined as the ratio of the power of the time-domain desired signal over the power of the time-domain noise at the reference microphone [7]. If $s'_1(n)$ is the target signal and $v'_1(n)$ is the noise during the same period, then the input SNR is

$$iSNR = 10 \log \frac{\sum_{n=1}^{N} s'^2_1(n)}{\sum_{n=1}^{N} v'^2_1(n)},$$

where N is the total number of time samples.

The output SNR has to be carefully defined, since there is no clear distinction of the filtered desired signal and the residual noise at the output signal of the beamformer in time. That is why it will be defined as the ratio of the power of the time-domain desired signal over the power of the difference of the target signal and the beamformer output in the time domain

$$oSNR = 10 \log \frac{\sum_{n=1}^{N} s'^2_1(n)}{\sum_{n=1}^{N} \left(s'_1(n) - \hat{s}'_1(n)\right)^2},$$

where $\hat{s}'_1(n)$ is the estimated target signal at the output of the beamformer.

6.2 Simulation Results

In this section, the beamforming results are plotted for the case of synchronized clocks in comparison to non-synchronized clocks (systems with clock offsets). Ideally, the performance of the beamformers has to be invariant to the existence of offsets.

6.2.1 Varying clock offsets, one target source and stationary interferers

First, the results for different clock offsets are given for one target source and the interfering sources producing white noise. It is reminded that in this case the RTFs are estimated through the GEVD.

6.2.1.1 Wiener filter

The results for the GEVD-based Wiener filter for different offsets are shown in Fig. 6.2. It can be seen that Fig. 6.2a and Fig. 6.2b show a small discrepancy between the synchronized and non-synchronized problems for high input SNR values. It appears as the filter performs better for positive clock offsets. This result is in absolute agreement with the analysis in Section 5.4: the method is sensitive to the window characteristics. It is impossible to obtain precisely the same SNR, as the signals are fragmented in different windows by the STFT, which are not shifted versions of each other. It is expected that the randomness of the offsets determines the exact performance, based on whether the specific offsets increase or decrease the correlation of signals within one time frame. In order to verify this, the simulations were also run with the smallest offset possible: only one microphone is allowed to have an offset of one sample, either positive or negative. This is illustrated in Figs. 6.2c and 6.2d. It is clear that in this case the performance is the same, with the output SNR values diverging only in the second decimal place.

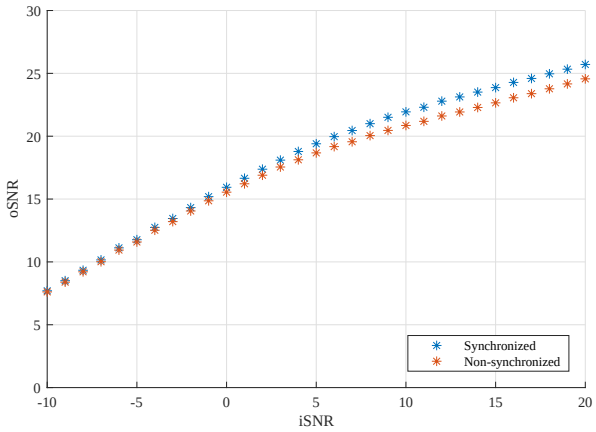
In order to support this interpretation even further, the simulations were also run with the same offsets as in Figs. 6.2a and 6.2b but with a longer STFT window of $48ms$. The results are shown in Figs. 6.3a and 6.3b. Increasing the window length leads to a slightly reduced difference between the synchronized and non-synchronized cases, as expected.

6.2.1.2 Trade-off filter with Low-Rank approximation of $\mathbf{R_X}$

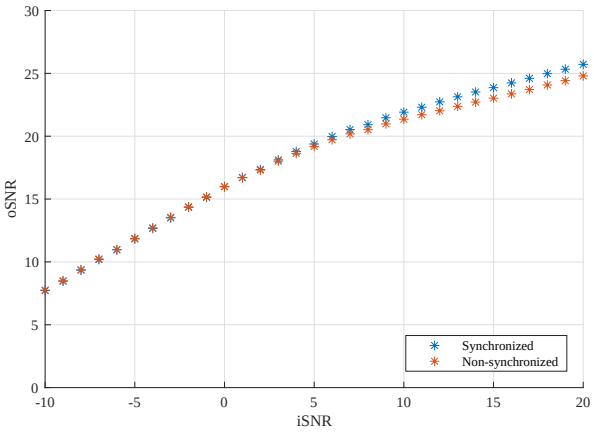
The performance of the trade-off filter when the CPSD matrix $\mathbf{R_X}$ is approximated by a lower rank matrix is shown in Fig. 6.4 for $R = 3$. This choice results in a beamformer with slightly worse noise reduction than the Wiener filter. This is predictable, as it allows a smaller distortion of the target signal. Regarding synchronization, this filter also behaves as the previous one and is almost entirely invariant to clock offsets.

6.2.2 Varying clock offsets, two target sources and stationary interferers

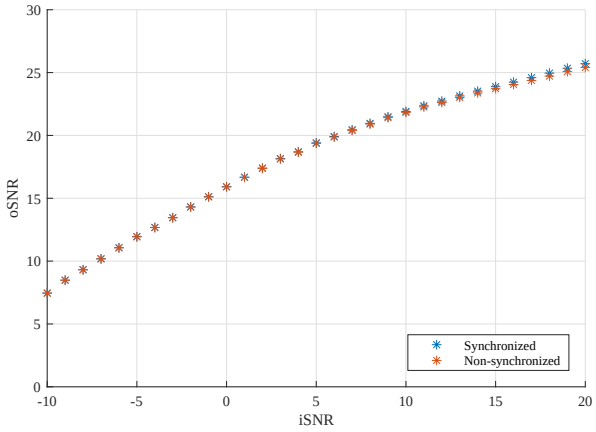
Here, the results are given for when there are $K = 2$ target sources. In this scenario, the beamformers concentrate on enhancing the sum of the desired sources.



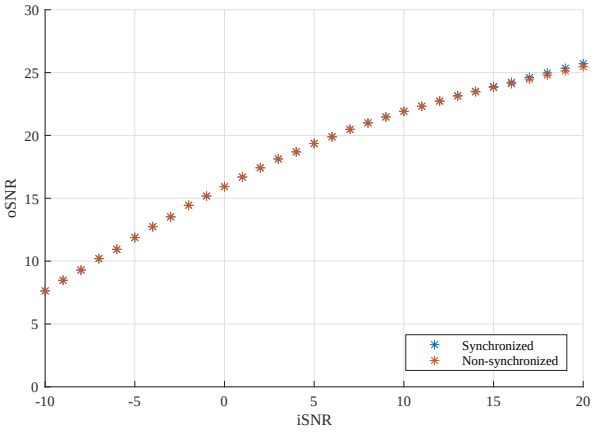
(a) Offsets equal to $[-10, -5, -6, -11]$.



(b) Offsets equal to $[10, 5, 6, 11]$.

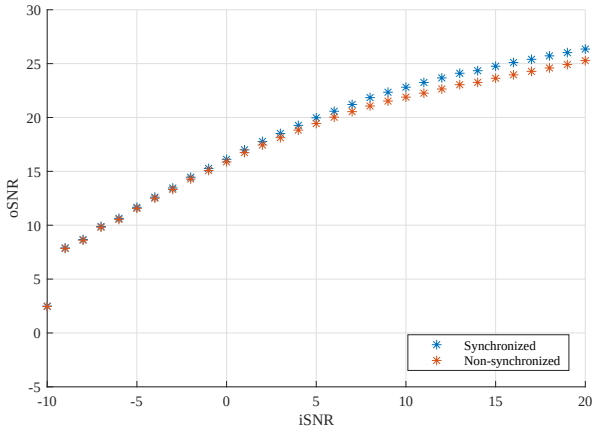


(c) Offsets equal to $[0, 0, 0, -1]$.

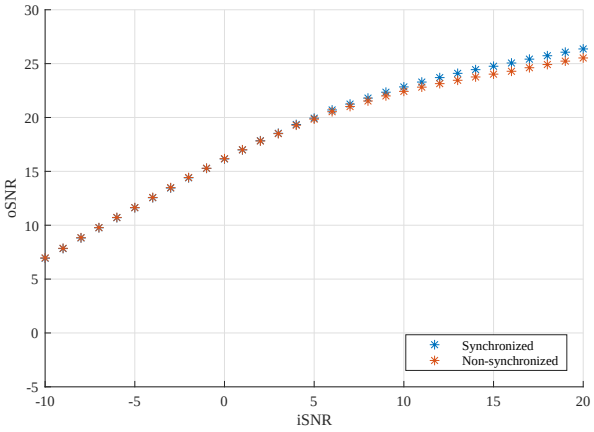


(d) Offsets equal to $[0, 0, 0, 1]$.

Figure 6.2: Wiener filter results for one target source and $32ms$ window.

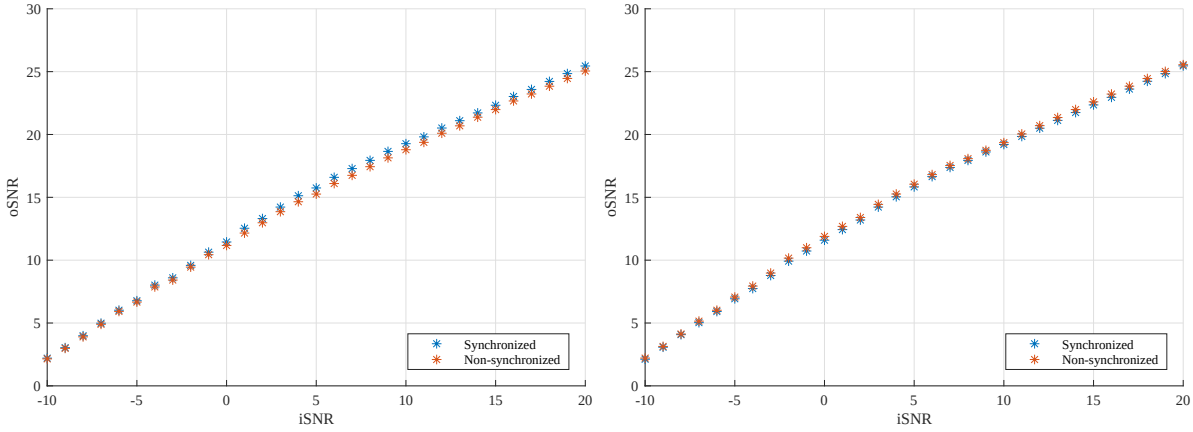


(a) Offsets equal to $[-10, -5, -6, -11]$.



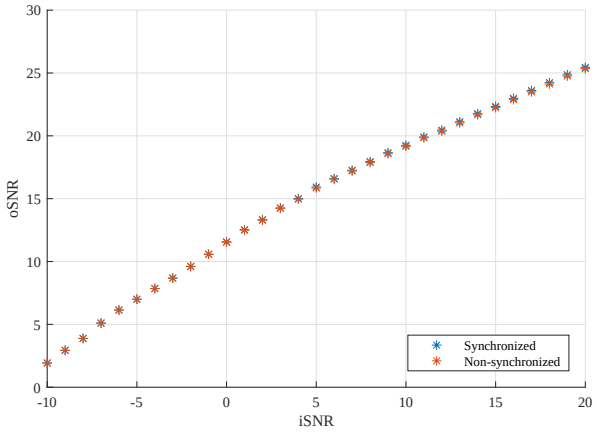
(b) Offsets equal to $[10, 5, 6, 11]$.

Figure 6.3: Wiener filter results for one target source and $48ms$ window.

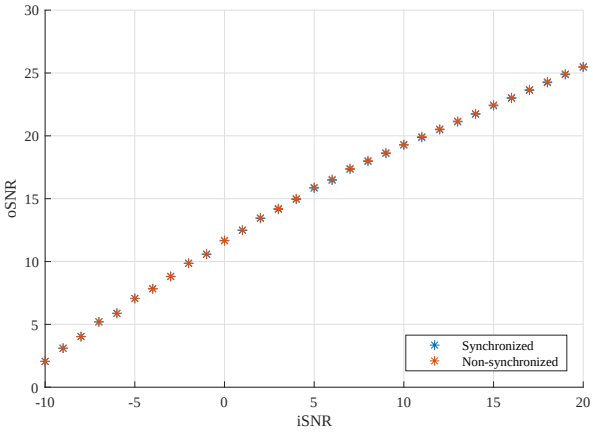


(a) Offsets equal to $[-10, -5, -6, -11]$.

(b) Offsets equal to $[10, 5, 6, 11]$.



(c) Offsets equal to $[0, 0, 0, -1]$.

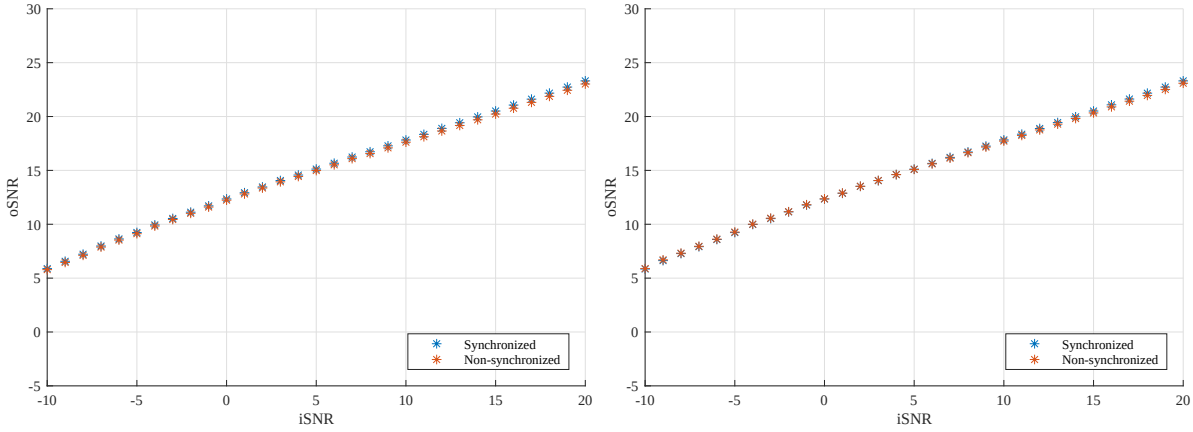


(d) Offsets equal to $[0, 0, 0, 1]$.

Figure 6.4: Trade-off filter with low-rank $\hat{\mathbf{R}}_{\mathbf{X}}$ results for one target source and $32ms$ window.

6.2.2.1 Wiener filter

The results for the GEVD-based Wiener filter for different offsets and two target sources are presented in Fig. 6.5. It is clear that the filters perform noise reduction and are invariant to clock offsets, as in the case of one target source. It is thus confirmed that the RTFs themselves need not be estimated: estimating their span is enough to perform noise suppression. This result can be generalized to a greater number of sources; however, one should not forget that when the number of target sources to be preserved increases, the amount of noise reduction is reduced (due to the trade-off of distortion and noise suppression). That is why the output SNR is lower here by $1-2dB$ compared to the case of one target source of Fig. 6.2. The number of microphones is an ultimate limitation to this problem.

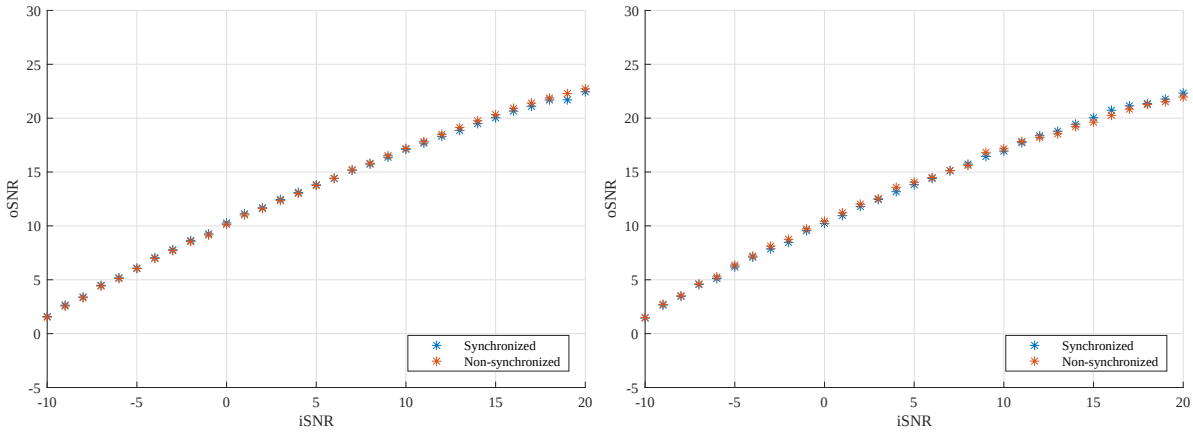


(a) Offsets equal to $[-10, -5, -6, -11]$. (b) Offsets equal to $[10, 5, 6, 11]$.

Figure 6.5: Wiener filter results for two target sources and $32ms$ window.

6.2.2.2 Trade-off filter with Low-Rank approximation of $\mathbf{R_X}$

The performance of the trade-off filter when the CPSD matrix $\mathbf{R_X}$ is approximated is shown in Fig. 6.6 for two target sources. The same remarks as for the Wiener filter hold.



(a) Offsets equal to $[-10, -5, -6, -11]$. (b) Offsets equal to $[10, 5, 6, 11]$.

Figure 6.6: Trade-off filter with low-rank $\hat{\mathbf{R}}_\mathbf{X}$ results for two target sources and $32ms$ window.

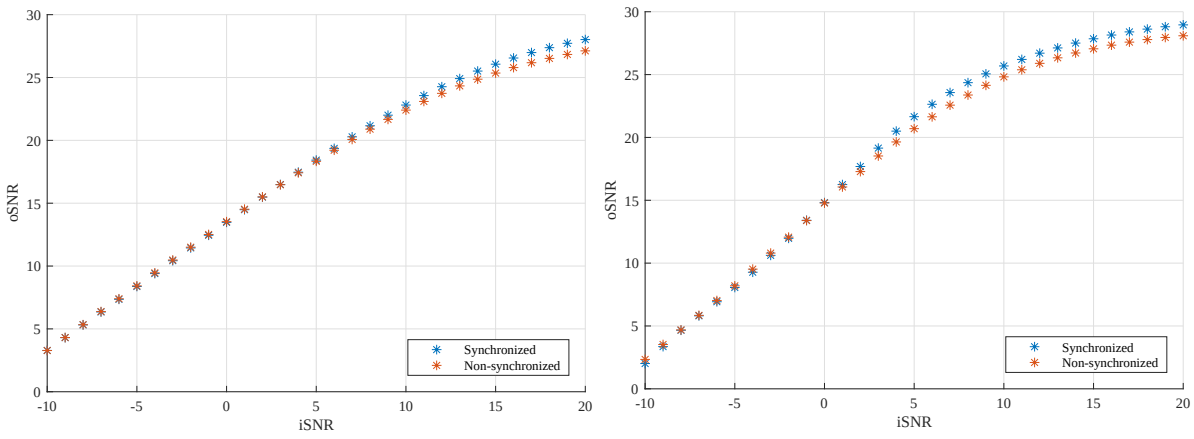
6.2.3 Varying clock offsets, one target source and non-stationary interferers

In this section, it is studied whether the same beamforming performance and the invariance to clock offsets are evident when all three interferers are non-stationary. The results are given in Fig. 6.7.

In both cases, it is evident that the filters behave similarly to the earlier cases and that they are suppressing the noise regardless of the non-stationarity of the interferers. This result is supported by [44], which demonstrated that the noise reduction performance of the GEVD-based filtering techniques is mainly dependent on the spatial characteristics of the noise sources and not on their temporal characteristics. Still, some sensitivity is expected with respect to the frequency content of the non-stationary interferers, a topic further discussed below.

In these tests, two of the three non-stationary signals were chosen to be speech signals, simulating a teleconferencing environment when non-target speakers are active at the same time with the target speakers. Interestingly, the plots suggest that the performance slightly improves for both filters when the interferers have this form.

This result is believed to be related to the essence of the frequency-domain signal subspace methods: the estimation of the span of the RTFs improves when the noise component in the specific frequency bin is reduced. This suggests that the noise suppression will be more effective in frequency bins where the target source components are dominant. In the frequency range over which typical speech signals have components, the speech subspace is well estimated. This is not true for frequency bins with high noise but low target signal content. When the interfering sources produce speech-like signals, their effect is well suppressed in the common frequency bins and the time signal, recovered by a form of averaging over frequencies, will have a good overall noise suppression. If the interfering signals have a frequency content much different than the target signals, it is expected that this result will not hold. For example, white noise signals have content over all frequencies and the noise reduction will not be as effective in the frequency bins where target speech is not present. Therefore, the GEVD-based algorithms appear to be affected by the frequency content of the noise and not by its non-stationarity.



(a) Wiener filter.

(b) Trade-off filter with low-rank $\hat{\mathbf{R}}_{\mathbf{X}}$.

Figure 6.7: Results for three non-stationary interferers and offsets equal to $[5, 6, -12, -3]$ for one target source and $32ms$ window.

6.2.4 Online implementation

It is interesting to evaluate the system performance with a real-time implementation of the algorithms. Fig. 6.8 refers to the case of one target source and 3 non-stationary interferers, where $\mathbf{R_Y}$ is estimated online and the Wiener filter is used. $\mathbf{R_V}$ is considered to have been already estimated during a period where the target source is inactive. It can be seen that the filters reduce the noise in clock-offset invariant way in this case, as well. The noise reduction is only slightly lower, mainly at low input SNR values.

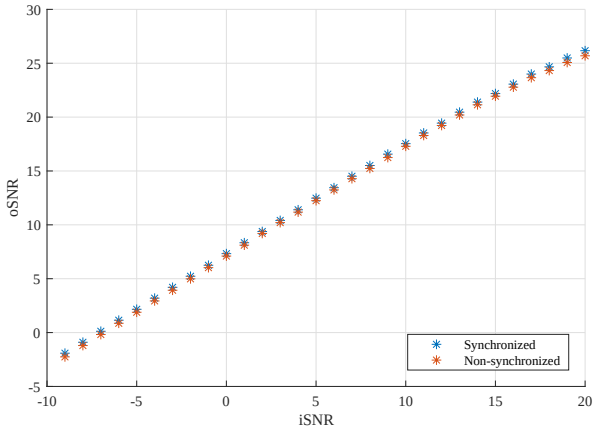
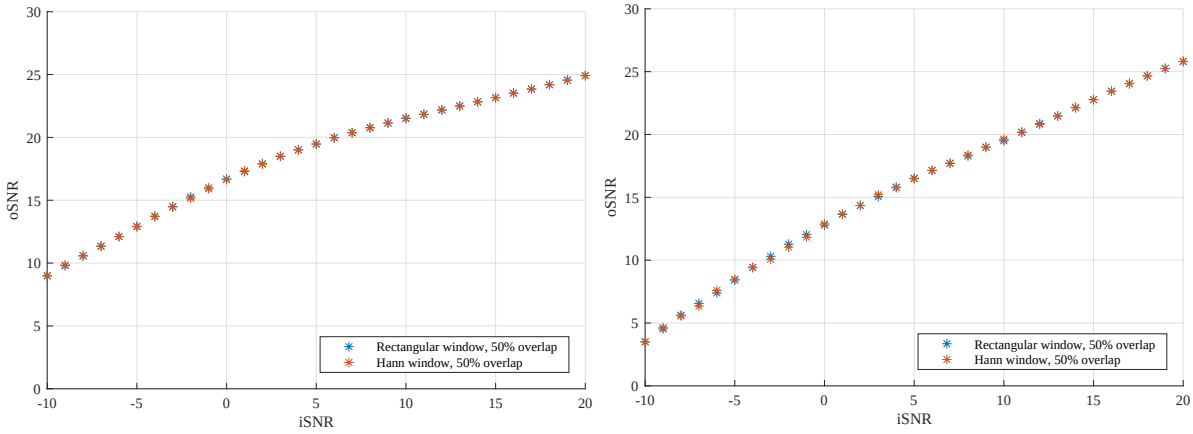


Figure 6.8: Wiener filter online implementation results for one target source, $32ms$ window and offsets equal to $[10, 5, 6, 11]$.

6.2.5 Window tests

A number of experiments was conducted to check the effect of the STFT window used for the estimation of the CPSD matrices on the beamforming process. The intention was to discover whether the span of the RTFs calculated through the GEVD carries some characteristics of the window. The STFT during beamforming is performed using a square root Hann window (see Appendix A.3). Therefore, it was interesting to check whether a CPSD matrix estimation method that uses the Hann window at 50% overlap would outperform the one that uses the rectangular window at 50% overlap. The results for the Wiener filter are given in Fig. 6.9a and for the trade-off filter with low rank $\hat{\mathbf{R}}_\mathbf{X}$ in Fig. 6.9b both for offsets equal to $[-8, -3, 2, 6]$. Therefore, these tests provide the indication that the choice of window when estimating the CPSD matrices does not affect the GEVD-based techniques. It appears that any of the two windows can be used with Welch's methodology of the modified periodogram to obtain a good estimate of the CPSD matrices.



(a) Wiener filter.

(b) Trade-off filter with low-rank $\hat{\mathbf{R}}_{\mathbf{X}}$.

Figure 6.9: Window tests for offsets equal to $[-8, -3, 2, 6]$, for one target source and $32ms$ window.

Conclusions and Recommendations

7

The distributed nature of WASNs raises important challenges for speech enhancement systems; a crucial aspect of this is the synchronization of the node clocks. In this thesis, it was explored how signal subspace methods, and particularly GEVD-based methods, cope with the issue of clock offsets.

Connected to the research question, in Chapter 5 the clock offsets were modeled and it was proved that blind beamforming techniques do not require an explicit estimation of and compensation for the clock offsets. For this, the CPSD matrices should be estimated using data of the processes as understood by the non-synchronized clocks. That is to say, when clock offsets are considered possible in a WASN, the calculation of the beamformers should not use prior information, for example the location-based RTFs, as noise suppression capabilities will be limited. This result was confirmed by simulations in Chapter 6, for stationary and non-stationary interfering sources. The simulations were limited in face of modeling inadequacies concerning room acoustics, and there could not be experiments for rooms with various shapes. However, generally the form of the RIR appears not to have any effect on the algorithm. GEVD-based beamforming techniques were found to work online as well, while the received process CPSD matrix is constantly updated. It should be mentioned, however, that in general these methods do not allow the estimation of the clock offset, in case this information is needed.

7.1 Discussion

Concluding this report, it is important to discuss the importance of all findings and some issues that could arise in practice. All in all, the GEVD provides a complete tool to support speech enhancement schemes. It deals with the clock offsets in the network and has other benefits, as well; mainly, that it does not require any knowledge about the system and only uses the available measurements. That is in contrast with conventional, fixed beamformers that are based on the knowledge of the RTFs, or, equivalently, of the source and sensor locations.

In the case of one target source, the GEVD can be used to estimate the target RTFs. This is also possible in a setting with multiple target sources, provided that for each of them there is at least one time segment during which it is the only active source.

More importantly, for beamforming purposes that involve more than one target sources, if GEVD-based methods are used, there is no need to know or explicitly estimate the RTFs. It suffices to estimate the span where the RTFs lie, which is delivered by the GEVD.

Furthermore, the GEVD provides the basis for a low-rank approximation of the target CPSD matrix, as an alternative to the difference of sample covariance matrices. In Chapter 6 the performance of a beamformer using this approximation was shown.

Regarding the clock offset problem, blind techniques based on the GEVD deal with the offsets internally, in a transparent way to the user, and it is not required to manually synchronize the sensors or to rearrange them spatially, in order to tackle the offsets. This outcome holds in both offline and online implementations. The latter exposes the superiority of this approach compared to methods based on maximum correlation of channel signals: they require entire data records to work, whilst they may not be as accurate.

Admittedly, the clock offsets cannot be estimated by the use of the GEVD as described. However, the offsets can be explicitly calculated in the case of one target source, by estimating the offset-affected RTFs using the GEVD and comparing them to the location RTFs which do not include clock offsets, should they be known.

Contemplating the implementation of the algorithm on a real system, it should be stressed that the clock offsets are arbitrary and unbounded, since the nodes are fully independent. All the tests for this thesis considered offsets that are well within one frame. That is why, in the most general case, when no prior information about the sensor offsets is available, the signals have to be coarsely aligned so that the processing can take place in time frames that are correlated for the channels. For an online application, this can be done using side information that might be available in the network, for example by exchanging frame indices or by broadcasting an acoustic signature upon the start of operation [26]. For an offline application it is fitting to perform this rough alignment based on maximizing the channel correlation for the entire signals. The performance of the system is expected to be challenged for sensors placed far from each other, as this alignment will not be a trivial task and traditional array techniques will not work. In this work, the sensors were not placed closely together and in the far field of sources, but they were still not placed in arbitrary locations in the room.

Finally, the signal subspace methods are connected to the general issue of the CPSD matrices estimation and are, therefore, sensitive to estimation inaccuracies. Especially for non-stationary signals, a number of time frames is needed to estimate the signal statistics. In this study, it was assumed that a VAD is available in the system and that a long period of only noise is available for the estimation, which might not be the case in reality. For the tests, it is interesting to point out that the low-rank approximation delivered good beamforming performance even if the number of sources was not considered known, simply but doing away with the eigenvectors with negative eigenvalues. Last but not least, this study and implementation accepted the general limitation of sources static in space, which might not hold in a teleconferencing application as the speakers often move in the room. Therefore, it would be useful to investigate how this element could be incorporated in the model.

7.2 Future Directions

The following ideas, which build upon the work presented in this thesis, may be useful for possible extensions and research topics.

It is suggested that future research be mainly directed to the distributed implementation of the algorithm. In this work, both the CPSD matrices, as sample covariance matrices, and the GEVD were evaluated at a central processor that has access to all the sensor data. The next step towards an actual implementation of this method is to solve the respective problems in a distributed way, possible through the primal-dual method of multipliers (PDMM), as researched in [41].

Indisputably, the area of signal subspace methods, and the GEVD technique in particular, are closely tied to the estimation of the CPSD matrices of processes. In this thesis, sample covariance matrices were used and it would be interesting to test how other estimation methods perform, such as a recursive exponential smoothing. Apart from the theoretical interest, this idea is practically relevant for cases when the clock offsets are not time-invariant or for when the sources are moving. It is important to also take the topology of the network into consideration, as large distances between the sensors can affect the performance on multiple levels.

Additionally, in this work, uniform networks were considered, where the clock skew problem is not normally present. A significant research direction is the inquiry into the application of GEVD-based beamforming techniques in networks where the devices are different and, therefore, clock skews are expected. Straightforwardly calculating the CPSD matrices using the entire data records will not yield proper estimates of the signal subspaces in this case, due to the asynchronous sampling instants. An idea for this is to adapt the CPSD matrices using a forgetting factor, as the skew problem can be perceived as a time-varying offset. It should be noted, however, that the skew problem requires explicit synchronization of the signals, as argued in [26].

On a different topic, practical experiments are also needed, where the mechanisms of communication could be tested and the real sensor offsets could be measured. A lot of literature work neglects this part. What is more, the performance measure throughout this work has been the SNR improvement from input to output. In a practical WASN, for example used for teleconferencing purposes, it is essential to test the aspect of speech intelligibility and how it is affected by the filtering operations and, more generally, develop performance measures related to the application. An example could be the intelligibility weighted signal distortion measure, described in [43].

As a final note, this thesis serves as proof of the significance of signal subspace methods. Given their promising character, their application in a field other than beamforming could be explored, for instance in a blind source separation (BSS) setting, as studied in [36].

Signals and Processes

A

A.1 Wide-Sense Stationary (WSS) Processes

As a disclaimer, the purpose of this section is rather to provide the relevant definitions for the subsequent analysis to be clear than to give strict mathematical descriptions for all terms involved.

With this in mind, the following definition of a stochastic process is given: "A stochastic (random) process is any collection, or *ensemble*, of random variables $\{X_n\}$ depending on time $n \in T$", where T is the time range involved. X_n or $X(n)$ symbolizes the random variable associated with fixed time t. Time n takes discrete values, such as $n = 0, 1, 2, \ldots$, thus T is a discrete set. The reader is invited to refer to [48–50] for more details on random variables and processes.

A stochastic process $\{X_n\}$ is usually denoted by X. By allowing a slight abuse of notation, it may also be denoted by X_n or $X(n)$. In practice, only a single realization x_n of this process is typically observed. Any of the possible realizations is a function of time n, as in $x_n = x(n)$, $n \in T$.

Now, on to some more problem-specific definitions.

For the real random process $\{X_n\}$, the *autocorrelation sequence* is given by

$$r_X(k, l) = \mathbb{E}[X(k)X(l)], \tag{A.1}$$

and the *autocovariance sequence* is given by

$$c_X(k, l) = \mathbb{E}[(X(k) - \mu_X(k))\,(X(l) - \mu_X(l))], \tag{A.2}$$

where $\mathbb{E}(\cdot)$ is the mathematical expectation operator and $\mu_X(k) = \mathbb{E}[X(k)]$ denotes the (deterministic) sequence known as the *mean* of the process. The autocovariance and the autocorrelation sequences provide information about the statistical relationship between two random variables that are derived from the same process [2], in this case $X(k)$ and $X(l)$.

A random process $\{X_n\}$ is said to be *wide-sense stationary* (WSS) if the following three conditions are satisfied [2]:

1. The mean of the process is a constant, independent of time, meaning $\mu_X(k) = \mu_X$.
2. The autocorrelation $r_X(k, l)$ depends only on the difference $k - l$ and not the time itself.
3. The *variance* of the process is finite, or $c_X(k, k) < \infty$.

The time difference $k - l$ is called the *lag*, and the second condition, in other words, requires $r_X(k, l) = r_X(k-l, 0)$. Permitting a slight abuse in notation, the zero argument is dropped and the autocorrelation is simply written as a function of the lag [2], as in

$$r_X(k, l) \equiv r_X(k - l). \tag{A.3}$$

A.2 Signal Analysis

In principle, a signal can have any functional form and it is possible to produce signals, such as sound waves, with extraordinary richness and complexity. Signal analysis is important, as a means of extracting information, drawing conclusions and commencing the processing of the signals. This analysis can take place in different domains and each of them has its advantages and disadvantages, as explained in the following sections.

A.2.1 Time Analysis

The most fundamental way of analyzing signals is in the *time domain*, where the analysis takes place with respect to the independent variable designated as time. By using techniques in this domain, it is possible to perform tasks such as peak detection, upsampling, downsampling and detection of signal inactivity periods. By careful inspection, it may also be possible to recognize periodicities in the signal.

A.2.2 Frequency Analysis

The frequency domain refers to the analysis of mathematical functions or signals with respect to *frequency*, rather than time. Frequency analysis or *spectral analysis* is a powerful mathematical tool and has developed greatly since its advent.

A signal can be converted between the time and frequency domains with a pair of mathematical operators called a *transform*. The transform relevant to this study is the *Fourier transform*, which converts a time function into a sum of sine waves of different frequencies, denoted *frequency components*, possibly infinite in number. After possible processing in the Fourier domain, the inverse Fourier transform is used to reconstruct the signal into a time function.

The two most common Fourier representations for discete-time signals are the DTFT for infinitely long data sequences, and the DFT for finite-duration sequences. More information on the DTFT can be found in [1].

A.2.3 Discrete Fourier Transform

For finite-duration discrete-time sequences there exists a different Fourier representation, referred to as the DFT. The DFT is itself a sequence rather than a function of a continuous variable. In addition to its theoretical importance as a Fourier representation of sequences, the DFT plays a central role in the implementation of digital signal

processing algorithms. Its discrete nature makes DFT calculations the most common practice for computers when extracting frequency information, since the DTFT cannot be computed numerically. This has been the motivation that led to efficient algorithms for the computation of the DFT, such as the FFT algorithm.

Consider a discrete-time signal $x(n)$ defined for $n = 0, \ldots, N-1$. The IDFT and DFTare written, respectively, as follows

$$x(n) = \frac{1}{N} \sum_{k=0}^{N-1} X(k) \mathrm{e}^{j\omega_k n}, \quad n = 0, \ldots, N-1 \tag{A.4}$$

and

$$X(\omega_k) = \sum_{n=0}^{N-1} x(n) \mathrm{e}^{-j\omega_k n}, \quad k = 0, \ldots, N-1, \tag{A.5}$$

where N is the length of the sequence $x(n)$ and the discrete frequency ω_k is defined as $\omega_k = 2\pi k/N$, $k = 0, 1, \ldots, N-1$. The variable ω_k has units of radians/sample, in which case it is called the *discrete normalized radian frequency variable*. A silent assumption here is that the sampling frequency is 1 Hz, or the sampling interval is 1 second. This is most typical in the digital signal processing literature.

An important note at this point is that the DFT delivers frequency components $X(\omega_k)$ equal in number to the number N of the signal samples in the time domain.

Since the frequency variable in the DFT representation is discrete, each spectral sample is associated with a small segment of the frequency continuum, rather than a point. More specifically, the kth spectral sample $X(\omega_k)$ is regarded as a measure of spectral amplitude over a range of frequencies, nominally $\omega_{k-1/2}$ to $\omega_{k+1/2}$. This range is usually called a *frequency bin*. The spectral index k is called the *bin number*.

An important property of the DFT is provided by the circular convolution theorem, which states that *circular convolution in the discrete-time domain becomes multiplication in the discrete-frequency domain.*, or

$$\mathrm{DFT}_{\omega_k}\{x(n) \circledast y(n)\} = \mathrm{DFT}_{\omega_k}\{x(n)\} \cdot \mathrm{DFT}_{\omega_k}\{y(n)\}, \tag{A.6}$$

where $x(n)$, $y(n)$ are discrete-time finite sequences, $\circledast$ denotes circular convolution and $\mathrm{DFT}_{\omega_k}\{\cdot\}$ stands for the DFT of a sequence expressed in terms of the frequency variable ω_k.

A.2.3.1 Some notes

A system is described in the Fourier domain by the *frequency response function*, which is the ratio of the output over the input. Consider an LTI discrete-time system with impulse response $h(n)$. Let $x(n)$ be a real-valued sequence that is a realization of a WSS discrete-time random process and is fed to the system as input.

By careful application of the DFT convolution theorem to Eq. (2.1), the *frequency response function* $H(\omega_k)$ of a system is obtained as

$$H(\omega_k) = \frac{Y(\omega_k)}{X(\omega_k)}. \tag{A.7}$$

In other words, it is the linear mapping of the Fourier transform of the input $X(\mathrm{e}^{j\omega})$ to the Fourier transform of the output $Y(\mathrm{e}^{j\omega})$. The frequency response function of a system is a special case of the *transfer function* of a system, which is defined in the Z-transform domain. More on this can be found in [3].

At this point, it should be noted that spectral analysis, as described above, is used for processing deterministic signals; however, it has a similar functionality in the study of random processes. What differentiates the approach is that a random process is a collection of signals, thus the Fourier transform cannot be applied to the process itself.

The *power spectrum* or *power spectral density* (PSD) of a random WSS process $\{X_n\}$ is the Fourier transform of its autocorrelation sequence $r_X(k)$, i.e.,

$$S_X(\mathrm{e}^{j\omega}) = \mathrm{DTFT}_\omega\{r_X\} = \sum_{k=-\infty}^{\infty} r_X(k)\mathrm{e}^{-j\omega k}. \tag{A.8}$$

where $\mathrm{DTFT}_\omega\{\cdot\}$ stands for the DTFT of a sequence expressed in terms of the frequency variable ω.

In the field of speech signal processing, it should be noted that most speech enhancement algorithms are performed in the spectral domain.

A.2.4 Time-Frequency Analysis

Looking back to the previous sections, a signal in time domain may be regarded as a representation with perfect time resolution and no frequency information. On the other hand, the Fourier transform of a signal may be considered to have perfect spectral resolution but no time information; that is because, in principle, frequency analysis, as demonstrated by Eq. (A.5), is conducted as an average over all time. As such, spectral analysis loses all chronological information and fails to convey when different events occur in the signal. This is not a problem for stationary signals, as their frequency components remain constant with time. However, it is a problem from non-stationary signals, such as speech and audio signals. In such a case it is important to investigate how the frequency content of a signal varies over time.

What is more, as revealed by Eq. (A.4), the computation of one frequency component necessitates the knowledge of the complete history of the signal. This can be challenging, especially in real-time applications. That is why it would be useful to divide the signal into segments, so that the processing can begin before the entire signal has been received.

The above are reasons that led to the development of methods in the *time-frequency domain*. Time-frequency representations provide both temporal and spectral information at the same time. Thus, they are particularly practical for the study of signals containing time-varying frequency components.

Given the number of different applications and some theoretical limitations that cannot be overcome, the problem of describing a signal in a joint time and frequency manner does not admit a unique answer. Numerous approaches and variations can be found in literature, each with its advantages and shortcomings. The method relevant to this study is described below.

A.3 Short-Time Fourier Transform (STFT)

The most typical time-frequency representation is obtained via the STFT. The STFT replaces the global Fourier analysis with a series of local analyses: the signal is localized by moving an observation window along the time axis, and applying the Fourier transform to obtain the frequency content of the signal for each position of the window. This transform provides a uniform resolution in time and frequency. Its usual mathematical definition is [51]

$$X_m(f) = \sum_{n=-\infty}^{\infty} x(n) w_A(n - mR) \mathrm{e}^{-j\omega n}, \tag{A.9}$$

where $x(n)$ is the input signal, $w_A(n)$ is the analysis window function of length M, R is the window hop size in samples and $X_m(f)$ is the DTFT of the windowed data centered around time mR, with m indicating the time frame. Often, it is $R < M$, in which case windows are overlapping.

While this definition of the STFT is useful for theoretical work, it does not provide a practical method of calculating it. In practice, the STFT is computed as

$$X_m(k) = \sum_{n=-N/2}^{N/2-1} x_A(n + mR) w(n) \mathrm{e}^{-j2\pi kn/N}. \tag{A.10}$$

In this form, the input signal is translated in time so that the data of time mR are moved to time 0 and then it is multiplied by the window $w(n)$ of length M. Afterwards, the DFT is performed in place of the DTFT. This sampling will not cause time aliasing if the number of frequency components is greater than the length of the windowed input data. This means then the DFT length should be $N \geq M$. It is typically a power of 2 to accelerate the FFT algorithm.

The time signal is reconstructed using the Inverse STFT (ISTFT), which is the IDFT of this sum, possibly including a synthesis window $w_S(n)$

$$x(n) = \sum_{m=-\infty}^{\infty} \sum_{k=0}^{N-1} X_m(k) w_S(n - mR) \mathrm{e}^{-j2\pi k(n-mR)/N}. \tag{A.11}$$

Throughout this thesis, it is assumed that $w_A(n)$ and $w_S(n)$ are real functions. Notice that the two windows are different in the general case. If the two windows fulfill the so-called *completeness condition*

$$\sum_{m=-\infty}^{\infty} w_A(n - mR)w_S(n - mR) = \text{constant}, \ \forall n \in \mathbb{Z}, \tag{A.12}$$

then a signal is guaranteed to be perfectly reconstructed from its STFT coefficients. However, for $R \leq M$ and for a given synthesis window $w_S(n)$, there might be an infinite number of solutions to Eq. (A.12) [52]; therefore, the choice of the two windows is generally not unique [53, 54].

In certain applications, e.g., when performing convolution using the STFT, the use of a synthesis window is skipped. Then, the completeness condition becomes

$$\sum_{m=-\infty}^{\infty} w_A(n - mR) = \text{constant}, \ \forall n \in \mathbb{Z}. \tag{A.13}$$

There are various windows that fulfill Eq. (A.13) for different amounts of overlap. Some commonly applied examples are the rectangular window at 0% overlap ($R = M$) and the Bartlett, Hann and Hamming windows at 50% overlap ($R = M/2$). Since the windows necessarily operate on a trade-off of the main lobe width and the side lobe level, different windows are preferred for different applications.

When a synthesis window is used, that is typically chosen to be the same as the analysis window, resulting in the constraint $\sum_{m=-\infty}^{\infty} w_A^2(n - mR) = \text{constant}, \forall n \in \mathbb{Z}$. This suggests a trivial way of constructing windows that satisfy Eq. (A.12) by taking the square root of any window that satisfies Eq. (A.13). This works for all non-negative windows and leads to windows such as the root-Hann or the root-Hamming.

The STFT evidently provides a series of benefits; however, it is worthwhile to mention that its structure carries essential restrictions that are typical for all Fourier methods, if used in a non-stationary context. The main limitation of the STFT is that it faces a trade-off between the temporal and the frequential resolution. The time resolution improves as the window becomes shorter; however, the frequency resolution degrades at the same rate since the Fourier analysis is confined to this short window. Conversely, a thinner frequency-resolution requires a longer window and, thus, yields a worse time resolution. This result is dictated by the Heisenberg uncertainty principle [55].

Glossary

B

RIR	room impulse response
ATF	acoustic transfer function
RTF	relative transfer function
A/D	analog-to-digital
WSS	wide-sense stationary
LTI	linear time-invariant
STFT	short-time Fourier transform
ISTFT	inverse STFT
DTFT	discrete-time Fourier transform
DFT	discrete Fourier transform
FFT	fast Fourier transform
IDFT	inverse DFT
CPSD	cross-power spectral density
GHEP	generalized Hermitian eigenvalue problem
GEVD	generalized eigenvalue decomposition
SVD	singular value decomposition
EVD	eigenvalue decomposition
FIR	finite impulse response
MSE	mean squared error
KKT	Karush-Kuhn-Tucker
SDW	signal-distortion weighted
MVDR	minimum variance distortionless response
VS	variable span

WSN	wireless sensor network
WASN	wireless acoustic sensor network
ICA	independent component analysis
UTC	Coordinated Universal Time
MTF	multiplicative transfer function
SNR	signal-to-noise ratio
SRO	sampling rate offset
DISS	delay in sampling start
TOA	time-of-arrival
VAD	voice activity detector

Bibliography

[1] A. V. Oppenheim and R. W. Schafer, *Discrete-Time Signal Processing*. Upper Saddle River, NJ, USA: Prentice Hall Press, 3rd ed., 2009.

[2] M. H. Hayes, *Statistical Digital Signal Processing and Modeling*. New York, NY, USA: John Wiley & Sons, Inc., 1st ed., 1996.

[3] B. Girod, R. Rabenstein, and A. Stenger, *Signals and Systems*. Wiley,, 2001.

[4] H. Kuttruff, *Room Acoustics*. CRC Press, 6th ed., 2016.

[5] T. Rossing, *Springer Handbook of Acoustics*. Springer Handbook of Acoustics, Springer New York, 2nd ed., 2007.

[6] X. Li, R. Horaud, L. Girin, and S. Gannot, "Local relative transfer function for sound source localization," in *2015 23rd European Signal Processing Conference (EUSIPCO)*, pp. 399–403, Aug 2015.

[7] I. Cohen, J. Benesty, and S. Gannot, *Speech Processing in Modern Communication: Challenges and Perspectives*. Springer Publishing Company, Incorporated, 2010.

[8] S. G. Tanyer and H. Ozer, "Voice activity detection in nonstationary noise," *IEEE Transactions on Speech and Audio Processing*, vol. 8, pp. 478–482, Jul 2000.

[9] S. M. Kay, *Fundamentals of Statistical Signal Processing: Estimation Theory*. Upper Saddle River, NJ, USA: Prentice-Hall, Inc., 1993.

[10] P. C. Loizou, *Speech Enhancement: Theory and Practice*. Boca Raton, FL, USA: CRC Press, Inc., 2nd ed., 2013.

[11] B. D. V. Veen and K. M. Buckley, "Beamforming: a versatile approach to spatial filtering," *IEEE ASSP Magazine*, vol. 5, pp. 4–24, April 1988.

[12] J. Benesty, J. Chen, Y. Huang, and I. Cohen, *Noise Reduction in Speech Processing*. Springer Publishing Company, Incorporated, 1st ed., 2009.

[13] S. Markovich-Golan, A. Bertrand, M. Moonen, and S. Gannot, "Optimal distributed minimum-variance beamforming approaches for speech enhancement in wireless acoustic sensor networks," *Signal Processing*, vol. 107, pp. 4 – 20, 2015. Special Issue on ad hoc microphone arrays and wireless acoustic sensor networks Special Issue on Fractional Signal Processing and Applications.

[14] Y. Zeng, *Distributed Speech Enhancement in Wireless Acoustic Sensor Networks*. PhD thesis, Delft University of Technology, Netherlands, 2015.

[15] R. Heusdens, G. Zhang, R. C. Hendriks, Y. Zeng, and W. B. Kleijn, "Distributed mvdr beamforming for (wireless) microphone networks using message passing," in *IWAENC 2012; International Workshop on Acoustic Signal Enhancement*, pp. 1–4, Sept 2012.

[16] Y. Zeng and R. C. Hendriks, "Distributed delay and sum beamformer for speech enhancement via randomized gossip," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 22, pp. 260–273, Jan 2014.

[17] A. A. Syed and J. Heidemann, "Time synchronization for high latency acoustic networks," in *Proceedings IEEE INFOCOM 2006. 25TH IEEE International Conference on Computer Communications*, pp. 1–12, April 2006.

[18] Y. C. Wu, Q. Chaudhari, and E. Serpedin, "Clock synchronization of wireless sensor networks," *IEEE Signal Processing Magazine*, vol. 28, pp. 124–138, Jan 2011.

[19] M. J. Whelan and K. D. Janoyan, "Design of a robust, high-rate wireless sensor network for static and dynamic structural monitoring," *Journal of Intelligent Material Systems and Structures*, vol. 20, no. 7, pp. 849–863, 2009.

[20] H. Kopetz and W. Ochsenreiter, "Clock synchronization in distributed real-time systems," *IEEE Transactions on Computers*, vol. C-36, pp. 933–940, Aug 1987.

[21] M. Horauer, K. Schossmaier, U. Schmid, T. Vienna, R. Hller, and N. Ker, "Psynutc-evaluation of a high-precision time synchronization prototype system for ethernet lans," Jul 2004.

[22] L. Schenato and F. Fiorentin, "Average timesynch: A consensus-based protocol for clock synchronization in wireless sensor networks," *Automatica*, vol. 47, no. 9, pp. 1878 – 1886, 2011.

[23] R. T. Rajan and A. J. van der Veen, "Joint ranging and clock synchronization for a wireless network," in *2011 4th IEEE International Workshop on Computational Advances in Multi-Sensor Adaptive Processing (CAMSAP)*, pp. 297–300, Dec 2011.

[24] S. Markovich-Golan, S. Gannot, and I. Cohen, "Blind sampling rate offset estimation and compensation in wireless acoustic sensor networks with application to beamforming," in *IWAENC 2012; International Workshop on Acoustic Signal Enhancement*, pp. 1–4, Sept 2012.

[25] J. Schmalenstroeer, P. Jebramcik, and R. Haeb-Umbach, "A combined hardware–software approach for acoustic sensor network synchronization," *Signal Processing*, vol. 107, pp. 171 – 184, February 2015.

[26] D. Cherkassky and S. Gannot, "Blind synchronization in wireless acoustic sensor networks," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 25, pp. 651–661, March 2017.

[27] M. H. Bahari, A. Bertrand, and M. Moonen, "Blind sampling rate offset estimation for wireless acoustic sensor networks through weighted least-squares coherence drift estimation," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 25, pp. 674–686, March 2017.

[28] N. Ono, H. Kohno, N. Ito, and S. Sagayama, "Blind alignment of asynchronously recorded signals for distributed microphone array," in *2009 IEEE Workshop on Applications of Signal Processing to Audio and Acoustics*, pp. 161–164, October 2009.

[29] J. Zhang, R. C. Hendriks, and R. Heusdens, "Structured total least squares based internal delay estimation for distributed microphone auto-localization," in *2016 IEEE International Workshop on Acoustic Signal Enhancement (IWAENC)*, pp. 1–5, Sept 2016.

[30] Z. Liu, "Sound source seperation with distributed microphone arrays in the presence of clocks synchronization errors," September 2008.

[31] J. P. Dmochowski, Z. Liu, and P. A. Chou, "Blind source separation in a distributed microphone meeting environment for improved teleconferencing," in *2008 IEEE International Conference on Acoustics, Speech and Signal Processing*, pp. 89–92, March 2008.

[32] S. Miyabe, N. Ono, and S. Makino, "Blind compensation of interchannel sampling frequency mismatch for ad hoc microphone array based on maximum likelihood estimation," *Signal Processing*, vol. 107, pp. 185 – 196, February 2015.

[33] J. Schmalenstroeer, P. Jebramcik, and R. Haeb-Umbach, "A gossiping approach to sampling clock synchronization in wireless acoustic sensor networks," in *2014 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 7575–7579, May 2014.

[34] S. Gannot, D. Burshtein, and E. Weinstein, "Signal enhancement using beamforming and nonstationarity with applications to speech," *IEEE Transactions on Signal Processing*, vol. 49, pp. 1614–1626, Aug 2001.

[35] Y. Ephraim and H. L. V. Trees, "A signal subspace approach for speech enhancement," *IEEE Transactions on Speech and Audio Processing*, vol. 3, pp. 251–266, Jul 1995.

[36] S. Markovich, S. Gannot, and I. Cohen, "Multichannel eigenspace beamforming in a reverberant noisy environment with multiple interfering speech signals," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 17, pp. 1071–1086, Aug 2009.

[37] J. R. Jensen, J. Benesty, and M. G. Christensen, "Noise reduction with optimal variable span linear filters," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 24, pp. 631–644, April 2016.

[38] G. H. Golub and C. F. Van Loan, *Matrix Computations (3rd Ed.)*. Baltimore, MD, USA: Johns Hopkins University Press, 1996.

[39] Z. Bai, J. Demmel, J. Dongarra, A. Ruhe, and H. van der Vorst, *Templates for the Solution of Algebraic Eigenvalue Problems*. Society for Industrial and Applied Mathematics, 2000.

[40] J. Benesty, J. Chen, and Y. Huang, *Microphone Array Signal Processing*, vol. 125. 04 2008.

[41] R. Heusdens, "Distributed optimal variable span linear filters," tech. rep., Delft University of Technology, January 2017.

[42] S. Doclo, A. Spriet, J. Wouters, and M. Moonen, "Frequency-domain criterion for the speech distortion weighted multichannel wiener filter for robust noise reduction," *Speech Commun.*, vol. 49, pp. 636–656, July 2007.

[43] R. Serizel, M. Moonen, B. V. Dijk, and J. Wouters, "Low-rank approximation based multichannel wiener filter algorithms for noise reduction with application in cochlear implants," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 22, pp. 785–799, April 2014.

[44] S. Doclo and M. Moonen, "Gsvd-based optimal filtering for single and multimicrophone speech enhancement," *IEEE Transactions on Signal Processing*, vol. 50, pp. 2230–2244, Sep 2002.

[45] A. Hassani, A. Bertrand, and M. Moonen, "Gevd-based low-rank approximation for distributed adaptive node-specific signal estimation in wireless sensor networks," *IEEE Transactions on Signal Processing*, vol. 64, pp. 2557–2572, May 2016.

[46] "RIR Generator." https://www.audiolabs-erlangen.de/fau/professor/habets/software/rir-generator. Accessed: 2018-06-30.

[47] J. B. Allen and D. A. Berkley, "Image method for efficiently simulating small-room acoustics," *The Journal of the Acoustical Society of America*, vol. 65, no. 4, pp. 943–950, 1979.

[48] J. Doob, *Stochastic processes*. Wiley publications in statistics, Wiley, 1990.

[49] M. Rosenblatt, *Random Processes*. Graduate Texts in Mathematics, Springer New York, 2nd ed., 1974.

[50] F. Klebaner, *Introduction to Stochastic Calculus with Applications*. Introduction to Stochastic Calculus with Applications, Imperial College Press, 2005.

[51] J. B. Allen and L. R. Rabiner, "A unified approach to short-time fourier analysis and synthesis," *Proceedings of the IEEE*, vol. 65, pp. 1558–1564, Nov 1977.

[52] Y. Avargel and I. Cohen, "System identification in the short-time fourier transform domain with crossband filtering," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 15, pp. 1305–1319, May 2007.

[53] J. Wexler and S. Raz, "Discrete gabor expansions," *Signal Processing*, vol. 21, no. 3, pp. 207 – 220, 1990.

[54] S. Qian and D. Chen, "Discrete gabor transform," *IEEE Transactions on Signal Processing*, vol. 41, pp. 2429–2438, Jul 1993.

[55] L. Cohen, *Time-frequency Analysis: Theory and Applications*. Upper Saddle River, NJ, USA: Prentice-Hall, Inc., 1995.