Carl von Ossietzky
Universität
Oldenburg

University of Applied Sciences
HOCHSCHULE
EMDEN·LEER

MIMO Convolutional Beamforming for Joint Dereverberation and Denoising

ℓ_p-Norm Reformulation of Weighted Power Minimization Distortionless Response (WPD) Beamforming

Author:
Henri Gode

Supervisors:
Prof. Dr. Simon Doclo
M.Sc. Marvin Tammen

Master Thesis

Engineering Physics (M.Sc.)

September 30, 2020

Acknowledgement

Firstly and most of all I want to thank and honor God, who indeed gave me the opportunity, abilities, strength and creativity to accomplish this work. Additionally I am very grateful for being part of the Signal Processing Group, which belongs to the Department of Medical Physics and Acoustics of the Carl von Ossietzky University Oldenburg. Also I want to appreciate the following persons for their support and encouragement during this work:

- **Prof. Dr. Simon Doclo** supported me during the whole process as my supervising professor. I could learn a lot from his teachings, the discussions and the recommendations of further literature. He is also the head of the signal processing group.

- **M.Sc. Marvin Tammen** supported me during the whole process as my second supervisor. He provided sample implementations and audio data for the experiments. In addition I got further understanding through discussions with him and his literature recommendations. He was also a help in terms of organisation.

- **B.Eng. Klaus Brümann** and **B.Eng. Wiebke Middelberg** for being great office mates, sharing programming codes and recommendations.

- My wife **Hanna Gode** who motivated and encouraged me throughout this time and provided extra time through overtaking some of my duties.

Abstract

Speech quality and intelligibility are important in every acoustic speech communication scenario. Since reverberation and noise have clear detrimental effects on the speech quality a need arises to develop algorithms, which perform dereverberation and noise reduction. Over the past years many algorithms have been proposed, which either perform dereverberation (e.g. weighted prediction error (WPE)) or noise reduction (e.g. minimum power distortionless response (MPDR) beamforming). To tackle both dereverberation and noise reduction, cascade systems have been proposed that consist of a multiple-input multiple-output (MIMO) dereverberation stage and a multiple-input single-output (MISO) beamforming stage, both optimized separately. In contrast to these cascade systems, recently the weighted power minimization distortionless response (WPD) algorithm has been proposed by Nakatani et al., which performs jointly optimized MIMO dereverberation and MISO beamforming also referred to as convolutional beamforming. This work aims at reformulating and modifying this WPD algorithm so that the sparsity of the cost function can be adjusted and a novel MIMO version can be derived. For this a mixed ℓ_p-norm is utilized as cost function. It is shown that the proposed MIMO-WPD is equivalent to a MISO version of WPD, whereby only the weight update is modified by an additional relative transfer function (RTF)-dependent term. In the experimental evaluation it is shown that the proposed MIMO-WPD significantly outperforms the MISO-WPD in terms of perceptual evaluation of speech quality (PESQ), frequency weighted segmetal SNR (FWSSNR) and cepstral distance (CD), which are widely accepted objective measures of speech quality. Additionally it is shown that the proposed MIMO-WPD needs fewer iterations for convergence, which corresponds to less computing cost.

Contents

List of Figures

List of Algorithms

Acronyms

A

ASR automatic speech recognition.

C

cATF convolutive acoustic transfer function.

CD cepstral distance.

CGG complex generalized Gaussian.

CW covariance whitening.

D

DAS delay-and-sum.

E

EVD eigenvalue decomposition.

F

FWSSNR frequency weighted segmetal SNR.

G

GSC generalized sidelobe canceler.

L

LH likelihood.

LS least-squares.

M

mATF multiplicative acoustic transfer function.

MCLP multi-channel linear prediction.

MIMO multiple-input multiple-output.

MISO multiple-input single-output.

MPDR minimum power distortionless response.

MVDR minimum variance distortionless response.

N

nLLH negative log-likelihood.

P

PCA principal component analysis.

PDF probability density function.

PESQ perceptual evaluation of speech quality.

PSD power spectral density.

R

RIR room impulse response.

RTF relative transfer function.

S

SCM sample covariance matrix.

SNR signal-to-noise ratio.

SPP speech presence probability.

STFT short time Fourier transform.

T

TVG time-varying complex Gaussian.

V

VAD voice activity detection.

W

wMPDR weighted minimum power distortionless response.

WPD weighted power minimization distortionless response.

WPE weighted prediction error.

Nomenclature

General Conventions

$e = \sum_{n=0}^{\infty} \frac{1}{n!} \approx 2.71828$ Euler's number; $e^{\bullet} = \exp(\bullet)$ is the exponential function.

$\mathbf{e}_M^{(m)} = \left[\underbrace{0, \dots, 0}_{m-1 \text{ zeros}}, \underbrace{1}_{m^{th}}, \underbrace{0, \dots, 0}_{M-m \text{ zeros}}\right]^T$ selection vector of length M containing only zeros except the m^{th} entry equals one.

$\mathbf{I}_M$ identity matrix of dimension $M \times M$.

$\mathbb{C}$ set of complex numbers.

$\mathbb{N}$ set of natural numbers.

$\mathbb{R}$ set of real numbers.

$\mathbb{R}_{>0}$ set of positive real numbers (without zero).

$\mathbb{R}_{\geq 0}$ set of positive real numbers (with zero).

Operators

$\bullet^*$ complex conjugate.

$\bullet^T$ non-conjugate matrix/vector transpose.

$\bullet^H$ conjugate matrix/vector transpose (hermitian).

$|\bullet| = \sqrt{\bullet\bullet^*}$ absolute value (applied on vector/matrix performs elementwise operation).

$\|\bullet\|_p = \sqrt[p]{\sum_{n=1}^{N} |\bullet_n|^p}$ ℓ_p vector norm.

$\|\bullet\|_{\text{Fro}} = \sqrt{\sum_{n_1=1}^{N_1} \sum_{n_2=1}^{N_2} |\bullet_{n_1,n_2}|^2}$ Frobenius norm of a $N_1 \times N_2$ matrix.

$\eta = \frac{\|\bullet^{(i)} - \bullet^{(i-1)}\|_{\{2,\text{Fro}\}}}{\|\bullet^{(i-1)}\|_{\{2,\text{Fro}\}}}$ relative convergence criterion.

$\mathbb{E}[\bullet]$ expectation operator.

$\Gamma(\bullet)$ gamma function.

$\mathfrak{L}(\bullet, \boldsymbol{\alpha}) = f(\bullet) + g(\bullet)\boldsymbol{\alpha}$ Lagrangian function used for constraint optimization whereby $f(\bullet)$ denotes the function to be optimized and $g(\bullet)$ the constrained.

$\boldsymbol{\alpha} \in \mathbb{C}^M$ auxiliary parameter vector for Lagrangian constraint optimization.

$\log(\bullet)$ natural logarithm (base $= e$).

$\texttt{trace}(\bullet)$ trace operator.

Functions and Algorithms

$\texttt{build}(\bullet)$ function / algorithm that builds a convolution matrix according to the algorithm (WPE or WPD) using the noisy input signal $\mathbf{Y}_k$, the prediction delay τ_k and the prediction filter length L_k.

$\texttt{cw}(\bullet)$ function / algorithm that estimates the RTF from the noisy and noise covariance matrix via covariance whitening.

$\texttt{diagMat}(\bullet)$ function that constructs a diagonal matrix, wherbey the entries of the diagonal correspond to the input vector.

$\texttt{matSqrt}(\bullet)$ function / algorithm that calculates an arbitrary matrix square root of the input matrix.

$\texttt{maxEigVec}(\bullet)$ function / algorithm that calculates the eigenvector corresponding to the maximal eigenvalue of the input matrix.

$\texttt{spp}(\bullet)$ function / algorithm that estimates an SPP mask of the input signal.

Indices and Total Numbers

$i \in \mathbb{N}$ index of iteration.

$K \in \mathbb{N}$ number of frequency subbands.

$k \in \mathbb{N}$ frequency subband index.

$l \in \mathbb{N}$ filter tap index.

$M \in \mathbb{N}$ number of microphones.

$m \in \mathbb{N}$ microphone index.

$T \in \mathbb{N}$ number of time frames.

$t \in \mathbb{N}$ time frame index.

Parameters

$\varepsilon \in \mathbb{R}_{>0}$ regularization constant (default $\varepsilon = 1 \times 10^{-8}$).

$\eta_c \in \mathbb{R}_{>0}$ convergence criterion of the alternating optimization (tolerance).

$I_{max} \in \mathbb{N}$ maximal number of iterations of the alternating optimization.

$L_k \in \mathbb{N}$ frequency dependent prediction filter length.

$p \in]0, 2]$ shape parameter determining sparsity of the ℓ_p-norm cost function.

$\tau_k \in \mathbb{N}$ frequency dependent prediction delay.

Signals (STFT-domain)

$s_{k,t} \in \mathbb{C}$ single-channel clean speech signal.

$y_{k,t}^{(m)} \in \mathbb{C}$ single-channel noisy microphone signal.

$\mathbf{y}_k^{(m)} = \left[y_{k,1}^{(m)}, y_{k,2}^{(m)}, \ldots, y_{k,T}^{(m)}\right] \in \mathbb{C}^{1 \times T}$ batch-vector of single-channel noisy microphone signal.

$\mathbf{y}_{k,t} = \left[y_{k,t}^{(1)}, y_{k,t}^{(2)}, \ldots, y_{k,t}^{(M)}\right]^T \in \mathbb{C}^M$ multi-channel noisy microphone signal.

$\mathbf{Y}_k = [\mathbf{y}_{k,1}, \mathbf{y}_{k,2}, \ldots, \mathbf{y}_{k,T}] \in \mathbb{C}^{M \times T}$ batch-matrix of multi-channel noisy microphone signal.

$\tilde{\mathbf{y}}_{k,t} = \left[\mathbf{y}_{k,t-\tau_k}^T, \mathbf{y}_{k,t-\tau_k-1}^T, \mathbf{y}_{k,t-\tau_k-2}^T, \ldots, \mathbf{y}_{k,t-L_k+1}^T\right]^T \in \mathbb{C}^{M(L_k-\tau_k)}$ multi-channel multi-frame stacked noisy microphone signal vector (only past frames corresponding to late reverberation).

$\tilde{\mathbf{Y}}_k = [\tilde{\mathbf{y}}_{k,1}, \tilde{\mathbf{y}}_{k,2}, \ldots, \tilde{\mathbf{y}}_{k,T}] \in \mathbb{C}^{M(L_k-\tau_k)\times T}$ batch-matrix of multi-channel multi-frame stacked noisy microphone signal (only past frames corresponding to late reverberation).

$\bar{\mathbf{y}}_{k,t} = \left[\mathbf{y}_{k,t}^T, \tilde{\mathbf{y}}_{k,t}^T\right]^T \in \mathbb{C}^{M(L_k-\tau_k+1)}$ multi-channel multi-frame stacked noisy microphone signal vector (current frame and past frames corresponding to late reverberation).

$\bar{\mathbf{Y}}_k = [\bar{\mathbf{y}}_{k,1}, \bar{\mathbf{y}}_{k,2}, \ldots, \bar{\mathbf{y}}_{k,T}] \in \mathbb{C}^{M(L_k-\tau_k+1)\times T}$ batch-matrix of multi-channel multi-frame stacked noisy microphone signal (current frame and past frames corresponding to late reverberation).

$x_{k,t}^{(m)} \in \mathbb{C}$ single-channel reverberant speech signal.

$\mathbf{x}_{k,t} = \left[x_{k,t}^{(1)}, x_{k,t}^{(2)}, \ldots, x_{k,t}^{(M)}\right]^T \in \mathbb{C}^M$ multi-channel reverberant speech signal.

$\mathbf{X}_k = [\mathbf{x}_{k,1}, \mathbf{x}_{k,2}, \ldots, \mathbf{x}_{k,T}] \in \mathbb{C}^{M\times T}$ batch-matrix of multi-channel reverberant speech signal.

$d_{k,t}^{(m)} \in \mathbb{C}$ single-channel desired signal.

$\mathbf{d}_k^{(m)} = \left[d_{k,1}^{(m)}, d_{k,2}^{(m)}, \ldots, d_{k,T}^{(m)}\right] \in \mathbb{C}^{1\times T}$ batch-vector of multi-channel desired signal.

$\mathbf{d}_{k,t} = \left[d_{k,t}^{(1)}, d_{k,t}^{(2)}, \ldots, d_{k,t}^{(M)}\right]^T \in \mathbb{C}^M$ multi-channel desired signal.

$\mathbf{D}_k = [\mathbf{d}_{k,1}, \mathbf{d}_{k,2}, \ldots, \mathbf{d}_{k,T}] \in \mathbb{C}^{M\times T}$ batch-matrix of multi-channel desired signal.

$\breve{d}_{k,t}^{(m)} \in \mathbb{C}$ single-channel dereverberated signal.

$\breve{\mathbf{d}}_k^{(m)} = \left[\breve{d}_{k,1}^{(m)}, \breve{d}_{k,2}^{(m)}, \ldots, \breve{d}_{k,T}^{(m)}\right] \in \mathbb{C}^{1\times T}$ batch-vector of multi-channel dereverberated signal.

$\breve{\mathbf{d}}_{k,t} = \left[\breve{d}_{k,t}^{(1)}, \breve{d}_{k,t}^{(2)}, \ldots, \breve{d}_{k,t}^{(M)}\right]^T \in \mathbb{C}^M$ multi-channel dereverberated signal.

$\breve{\mathbf{D}}_k = \left[\breve{\mathbf{d}}_{k,1}, \breve{\mathbf{d}}_{k,2}, \ldots, \breve{\mathbf{d}}_{k,T}\right] \in \mathbb{C}^{M\times T}$ batch-matrix of multi-channel dereverberated signal.

$r_{k,t}^{(m)} \in \mathbb{C}$ single-channel late reverberant tail signal.

$\mathbf{r}_{k,t} = \left[r_{k,t}^{(1)}, r_{k,t}^{(2)}, \ldots, r_{k,t}^{(M)}\right]^T \in \mathbb{C}^M$ multi-channel late reverberant tail signal.

$n_{k,t}^{(m)} \in \mathbb{C}$ single-channel additive noise signal.

$\mathbf{n}_{k,t} = \left[n_{k,t}^{(1)}, n_{k,t}^{(2)}, \dots, n_{k,t}^{(M)}\right]^T \in \mathbb{C}^M$ multi-channel additive noise signal.

$w_{k,t}^{(m)} \in \mathbb{C}$ single-channel whitened microphone signal.

$\mathbf{w}_{k,t} = \left[w_{k,t}^{(1)}, w_{k,t}^{(2)}, \dots, w_{k,t}^{(M)}\right]^T \in \mathbb{C}^M$ multi-channel whitened microphone signal.

$\mathbf{W}_k = [\mathbf{w}_{k,1}, \mathbf{w}_{k,2}, \dots, \mathbf{w}_{k,T}] \in \mathbb{C}^{M \times T}$ batch-matrix of multi-channel whitened microphone signal.

$z_{k,t}^{(m)} \in \mathbb{C}$ single-channel beamformed signal.

$\mathbf{z}_k^{(m)} = \left[z_{k,1}^{(m)}, z_{k,2}^{(m)}, \dots, z_{k,T}^{(m)}\right] \in \mathbb{C}^{1 \times T}$ batch-vector of single-channel beamformed signal.

$\mathbf{z}_{k,t} = \left[z_{k,t}^{(1)}, z_{k,t}^{(2)}, \dots, z_{k,t}^{(M)}\right]^T \in \mathbb{C}^M$ multi-channel beamformed signal.

$\mathbf{Z}_k = [\mathbf{z}_{k,1}, \mathbf{z}_{k,2}, \dots, \mathbf{z}_{k,T}] \in \mathbb{C}^{M \times T}$ batch-matrix of multi-channel beamformed signal.

Signal Statistics

$\phi_{s,k} = \mathbb{E}\left[s_{k,t} s_{k,t}^*\right] \approx \frac{1}{T} \sum_{t=1}^T |s_{k,t}|^2 \in \mathbb{R}_{\geq 0}$ speech PSD.

$\mathbf{R}_{y,k} = \mathbb{E}\left[\mathbf{y}_{k,t} \mathbf{y}_{k,t}^H\right] \approx \frac{1}{T} \mathbf{Y}_k \mathbf{Y}_k^H = \frac{1}{T} \sum_{t=1}^T \mathbf{y}_{k,t} \mathbf{y}_{k,t}^H \in \mathbb{C}^{M \times M}$ noisy covariance matrix.

$\mathbf{R}_{y\lambda,k} = \mathbb{E}\left[\frac{\mathbf{y}_{k,t} \mathbf{y}_{k,t}^H}{\lambda_{k,t}}\right] \approx \frac{1}{T} \mathbf{Y}_k \mathbf{\Lambda}_k^{-1} \mathbf{Y}_k^H = \frac{1}{T} \sum_{t=1}^T \frac{\mathbf{y}_{k,t} \mathbf{y}_{k,t}^H}{\lambda_{k,t}} \in \mathbb{C}^{M \times M}$ power weighted noisy covariance matrix.

$\tilde{\mathbf{R}}_{\tilde{y}\lambda,k} = \mathbb{E}\left[\frac{\tilde{\mathbf{y}}_{k,t} \tilde{\mathbf{y}}_{k,t}^H}{\lambda_{k,t}}\right] \approx \frac{1}{T} \tilde{\mathbf{Y}}_k \mathbf{\Lambda}_k^{-1} \tilde{\mathbf{Y}}_k^H = \frac{1}{T} \sum_{t=1}^T \frac{\tilde{\mathbf{y}}_{k,t} \tilde{\mathbf{y}}_{k,t}^H}{\lambda_{k,t}} \in \mathbb{C}^{M(L_k - \tau_k) \times M(L_k - \tau_k)}$ power weighted noisy covariance matrix of past frames corresponding to late reverberation (batch).

$\mathbf{p}_{y\lambda,k}^{(m)} = \mathbb{E}\left[\frac{\tilde{\mathbf{y}}_{k,t} y_{k,t}^{(m)*}}{\lambda_{k,t}}\right] \approx \frac{1}{T} \tilde{\mathbf{Y}}_k \mathbf{\Lambda}_k^{-1} \mathbf{y}_k^{(m)H} = \frac{1}{T} \sum_{t=1}^T \frac{\tilde{\mathbf{y}}_{k,t} y_{k,t}^{(m)*}}{\lambda_{k,t}} \in \mathbb{C}^{M(L_k - \tau_k)}$ power weighted noisy cross-covariance vector of current frame at the m^{th} channel with past frames corresponding to late reverberation.

$\mathbf{P}_{y\lambda,k} = \mathbb{E}\left[\frac{\tilde{\mathbf{y}}_{k,t} \mathbf{y}_{k,t}^H}{\lambda_{k,t}}\right] = \left[\mathbf{p}_{y,k}^{(1)}, \mathbf{p}_{y,k}^{(2)}, \dots, \mathbf{p}_{y,k}^{(M)}\right] \approx \frac{1}{T} \tilde{\mathbf{Y}}_k \mathbf{\Lambda}_k^{-1} \mathbf{Y}_k^H = \frac{1}{T} \sum_{t=1}^T \frac{\tilde{\mathbf{y}}_{k,t} \mathbf{y}_{k,t}^H}{\lambda_{k,t}} \in \mathbb{C}^{M(L_k - \tau_k) \times M}$ power weighted noisy cross-covariance matrix of current frame with past frames corresponding to late reverberation.

$\bar{\mathbf{R}}_{\bar{y}\lambda,k} = \mathbb{E}\left[\frac{\bar{\mathbf{y}}_{k,t}\bar{\mathbf{y}}_{k,t}^H}{\lambda_{k,t}}\right] = \begin{bmatrix}\mathbf{R}_{y\lambda,k} & \mathbf{P}_{y\lambda,k}^H \\ \mathbf{P}_{y\lambda,k} & \tilde{\mathbf{R}}_{\tilde{y}\lambda,k}\end{bmatrix} \approx \frac{1}{T}\bar{\mathbf{Y}}_k\boldsymbol{\Lambda}_k^{-1}\bar{\mathbf{Y}}_k^H = \frac{1}{T}\sum_{t=1}^{T}\frac{\bar{\mathbf{y}}_{k,t}\bar{\mathbf{y}}_{k,t}^H}{\lambda_{k,t}} \in \mathbb{C}^{M(L_k-\tau_k+1)\times M(L_k-\tau_k+1)}$ power weighted noisy covariance matrix of current frame and past frames corresponding to late reverberation.

$\mathbf{R}_{d,k} = \mathbb{E}\left[\mathbf{d}_{k,t}\mathbf{d}_{k,t}^H\right] \approx \frac{1}{T}\sum_{t=1}^{T}\mathbf{d}_{k,t}\mathbf{d}_{k,t}^H \in \mathbb{C}^{M\times M}$ desired speech covariance matrix.

$\mathbf{R}_{\breve{d},k} = \mathbb{E}\left[\breve{\mathbf{d}}_{k,t}\breve{\mathbf{d}}_{k,t}^H\right] \approx \frac{1}{T}\breve{\mathbf{D}}_k\breve{\mathbf{D}}_k^H = \frac{1}{T}\sum_{t=1}^{T}\breve{\mathbf{d}}_{k,t}\breve{\mathbf{d}}_{k,t}^H \in \mathbb{C}^{M\times M}$ dereverberated covariance matrix.

$\mathbf{R}_{\breve{d}\lambda,k} = \mathbb{E}\left[\frac{\breve{\mathbf{d}}_{k,t}\breve{\mathbf{d}}_{k,t}^H}{\lambda_{k,t}}\right] \approx \frac{1}{T}\breve{\mathbf{D}}_k\boldsymbol{\Lambda}_k^{-1}\breve{\mathbf{D}}_k^H = \frac{1}{T}\sum_{t=1}^{T}\frac{\breve{\mathbf{d}}_{k,t}\breve{\mathbf{d}}_{k,t}^H}{\lambda_{k,t}} \in \mathbb{C}^{M\times M}$ power weighted dereverberated covariance matrix.

$\mathbf{R}_{n,k} = \mathbb{E}\left[\mathbf{n}_{k,t}\mathbf{n}_{k,t}^H\right] \approx \frac{1}{T}\sum_{t=1}^{T}\mathbf{n}_{k,t}\mathbf{n}_{k,t}^H \in \mathbb{C}^{M\times M}$ noise covariance matrix.

$\mathbf{R}_{w,k} = \mathbb{E}\left[\mathbf{w}_{k,t}\mathbf{w}_{k,t}^H\right] \approx \frac{1}{T}\sum_{t=1}^{T}\mathbf{w}_{k,t}\mathbf{w}_{k,t}^H \in \mathbb{C}^{M\times M}$ whitened noisy covariance matrix.

$\gamma_{w,k}^{(\max)} = \gamma_{w,k}^{(1)} \in \mathbb{R}$ maximal eigenvalue of whitened covariance matrix (eigenvalues are sorted, from largest to smallest).

$\boldsymbol{\Gamma}_{w,k} = \begin{bmatrix}\gamma_{w,k}^{(1)} & & & \mathbf{0} \\ & \gamma_{w,k}^{(2)} & & \\ & & \ddots & \\ \mathbf{0} & & & \gamma_{w,k}^{(M)}\end{bmatrix} \in \mathbb{C}^{M\times M}$ diagonal eigenvalue-matrix of whitened covariance matrix.

$\boldsymbol{\psi}_{w,k}^{(\max)} = \boldsymbol{\psi}_{w,k}^{(1)} \in \mathbb{C}^{M}$ eigenvector corresponding to maximal eigenvalue of whitened covariance matrix.

$\boldsymbol{\Psi}_{w,k} = \left[\boldsymbol{\psi}_{w,k}^{(1)}, \boldsymbol{\psi}_{w,k}^{(2)}, \ldots, \boldsymbol{\psi}_{w,k}^{(M)}\right] \in \mathbb{C}^{M\times M}$ eigenvector-matrix of whitened covariance matrix.

$\mathbf{R}_{z,k} = \mathbb{E}\left[\mathbf{z}_{k,t}\mathbf{z}_{k,t}^H\right] \approx \frac{1}{T}\sum_{t=1}^{T}\mathbf{z}_{k,t}\mathbf{z}_{k,t}^H \in \mathbb{C}^{M\times M}$ beamformed covariance matrix.

$\lambda_{k,t} \in \mathbb{R}_{>0}$ optimization weights, which depend on the power of the desired signal in the k^{th} frequency subband and the t^{th} time frame.

$\boldsymbol{\lambda}_k = [\lambda_{k,1}, \lambda_{k,2}, \ldots, \lambda_{k,T}] \in \mathbb{R}_{>0}^{1\times T}$ batch-vector of optimization weights, which depend on the power of the desired signal in the k^{th} frequency subband.

$\mathbf{\Lambda}_k = \mathtt{diagMat}\left(\boldsymbol{\lambda}_k\right) = \begin{bmatrix} \lambda_{k,1} & & & \mathbf{0} \\ & \lambda_{k,2} & & \\ & & \ddots & \\ \mathbf{0} & & & \lambda_{k,T} \end{bmatrix} \in \mathbb{R}_{>0}^{T\times T}$ batch-matrix of optimization weights, which depend on the power of the desired signal in the k^{th} frequency subband.

$\rho\left(d_{k,t}^{(m)}\right) \in [0,1]$ PDF of single-channel desired signal $d_{k,t}^{(m)}$.

$\mathbb{L}\left(\mathbf{d}_k^{(m)}\right) = \prod_{t=1}^{T} \rho\left(d_{k,t}^{(m)}\right) \in [0,1]$ LH function of batch-vector of single-channel desired signal $\mathbf{d}_k^{(m)}$.

$\mathcal{L}\left(\mathbf{d}_k^{(m)}\right) = -\log \mathbb{L}\left(\mathbf{d}_k^{(m)}\right) = -\sum_{t=1}^{T} \log \rho\left(d_{k,t}^{(m)}\right) \in \mathbb{R}_{\geq 0}$ nLLH function of batch vector of single-channel desired signal $\mathbf{d}_k^{(m)}$.

$\mathcal{C}\left(\mathbf{d}_k^{(m)}\right) \in \mathbb{R}$ cost function of batch-vector of single-channel desired signal $\mathbf{d}_k^{(m)}$.

$\beta \in \mathbb{R}_{>0}$ scale parameter of the CGG sparse prior.

$\mathbf{\Phi} \in \mathbb{R}^{M\times M}$ matrix to model group correlation structure within the mixed $\ell_{\mathbf{\Phi};2,p}$-norm.

$\sigma_{k,t} \in [0,1]$ speech presence probability of the k^{th} frequency subband and the t^{th} time frame.

$\boldsymbol{\sigma}_k = \left[\sigma_{k,1}, \sigma_{k,2}, \ldots, \sigma_{k,T}\right] \in [0,1]^{1\times T}$ batch-vector of speech presence probability in the k^{th} frequency subband.

$\mathbf{\Sigma}_k = \mathtt{diagMat}\left(\boldsymbol{\sigma}_k\right) = \begin{bmatrix} \sigma_{k,1} & & & \mathbf{0} \\ & \sigma_{k,2} & & \\ & & \ddots & \\ \mathbf{0} & & & \sigma_{k,T} \end{bmatrix} \in [0,1]^{T\times T}$ batch-matrix of speech presence probability in the k^{th} frequency subband.

Filters and Transfer Functions (STFT-domain)

$\mathbf{a}_{k,l} = \left[a_{k,l}^{(1)}, a_{k,l}^{(2)}, \ldots, a_{k,l}^{(M)}\right]^T \in \mathbb{C}^M$ multi-channel cATF coefficients (batch).

$v_k^{(m)} \in \mathbb{C}$ single-channel mATF (batch).

$\mathbf{v}_k = \left[v_k^{(1)}, v_k^{(2)}, \ldots, v_k^{(M)}\right]^T \in \mathbb{C}^M$ multi-channel mATF (batch).

$\tilde{\mathbf{v}}_k^{(m)} = \frac{\mathbf{v}_k}{v_k^{(m)}} = \frac{\left[v_k^{(1)}, v_k^{(2)}, \dots, v_k^{(M)}\right]^T}{v_k^{(m)}} \in \mathbb{C}^M$ single-channel RTF using channel m as reference (batch).

$\bar{\mathbf{v}}_k^{(m)} = \left[\tilde{\mathbf{v}}_k^{(m)},\ \underbrace{0, 0, \dots, 0}_{M(L_k - \tau_k)\ \text{zeros}}\right]^T \in \mathbb{C}^{M(L_k - \tau_k + 1)}$ single-channel RTF using channel m as reference (batch) extended with zeros for the past frames corresponding to late reverberation.

$\bar{\mathbf{v}}_k = \left[\mathbf{v}_k,\ \underbrace{0, 0, \dots, 0}_{M(L_k - \tau_k)\ \text{zeros}}\right]^T \in \mathbb{C}^{M(L_k - \tau_k + 1)}$ multi-channel mATF (batch) extended with zeros for the past frames corresponding to late reverberation.

$\tilde{\mathbf{g}}_k^{(m)} \in \mathbb{C}^{M(L_k - \tau_k)}$ single-channel reverberation filter vector using channel m as reference (batch).

$\bar{\mathbf{g}}_k^{(m)} = \begin{bmatrix} \mathbf{e}_M^{(m)} \\ -\tilde{\mathbf{g}}_k^{(m)} \end{bmatrix} \in \mathbb{C}^{M(L_k - \tau_k + 1)}$ single-channel dereverberation filter vector using channel m as reference (batch).

$\tilde{\mathbf{G}}_k = \left[\tilde{\mathbf{g}}_k^{(1)}, \tilde{\mathbf{g}}_k^{(2)}, \dots, \tilde{\mathbf{g}}_k^{(M)}\right] \in \mathbb{C}^{M(L_k - \tau_k) \times M}$ multi-channel reverberation filter matrix (batch).

$\bar{\mathbf{G}}_k = \begin{bmatrix} \mathbf{I}_M \\ -\tilde{\mathbf{G}}_k \end{bmatrix} = \left[\bar{\mathbf{g}}_k^{(1)}, \bar{\mathbf{g}}_k^{(2)}, \dots, \bar{\mathbf{g}}_k^{(M)}\right] \in \mathbb{C}^{M(L_k - \tau_k + 1) \times M}$ multi-channel dereverberation filter matrix (batch).

$\mathbf{q}_k \in \mathbb{C}^M$ single-channel general denoising filter vector (batch).

$\mathbf{q}_k^{(m)} \in \mathbb{C}^M$ single-channel normalized denoising filter vector using channel m as reference (batch).

$\mathbf{Q}_k = \left[\mathbf{q}_k^{(1)}, \mathbf{q}_k^{(2)}, \dots, \mathbf{q}_k^{(M)}\right] \in \mathbb{C}^{M \times M}$ multi-channel normalized denoising filter matrix using all M channels as references (batch).

$\bar{\mathbf{h}}_k^{(m)} = \bar{\mathbf{G}}_k \mathbf{q}_k^{(m)} \in \mathbb{C}^{M(L_k - \tau_k + 1)}$ single-channel normalized convolutional beamformer filter vector using channel m as reference (batch).

$\bar{\mathbf{H}}_k = \bar{\mathbf{G}}_k \mathbf{Q}_k = \left[\bar{\mathbf{h}}_k^{(1)}, \bar{\mathbf{h}}_k^{(2)}, \dots, \bar{\mathbf{h}}_k^{(M)}\right] \in \mathbb{C}^{M(L_k - \tau_k + 1) \times M}$ multi-channel normalized convolutional beamformer filter matrix using all M channels as references (batch).

Chapter 1

Introduction

This work mainly aims at extending and improving the state-of-the-art convolutional beamformer algorithm, performing unified dereverberation and denoising of acoustic speech signals, proposed by Nakatani et al. [1].

1.1 Motivation

One important aspect of our human life is the ability to communicate and interact with each another. Hereby acoustic speech communication plays a central role since we utter approximately 16000 words per day on average [2]. The intelligibility of the speech is essential but can often be a challenge in everyday situations. The main reason is that reverberation and noise can have strong detrimental effects on the speech quality especially for hearing impaired persons [3–5]. Some examples of these everyday situations are:

- A city center with traffic noise
- Large buildings (e.g. malls, churches) with long reverberation times
- Crowded events with interfering speakers

The aforementioned convolutional beamformer algorithm aims at improving the speech quality by performing dereverberation and denoising on these noisy and reverberant speech signals. The main applications, where combined dereverberation and denoising can have great benefits, are:

- Hearing aids for hearing impaired persons
- Conference telephony
- Smart speakers with automatic speech recognition (ASR) systems

Also digital signal processing is possible in all of the mentioned applications.

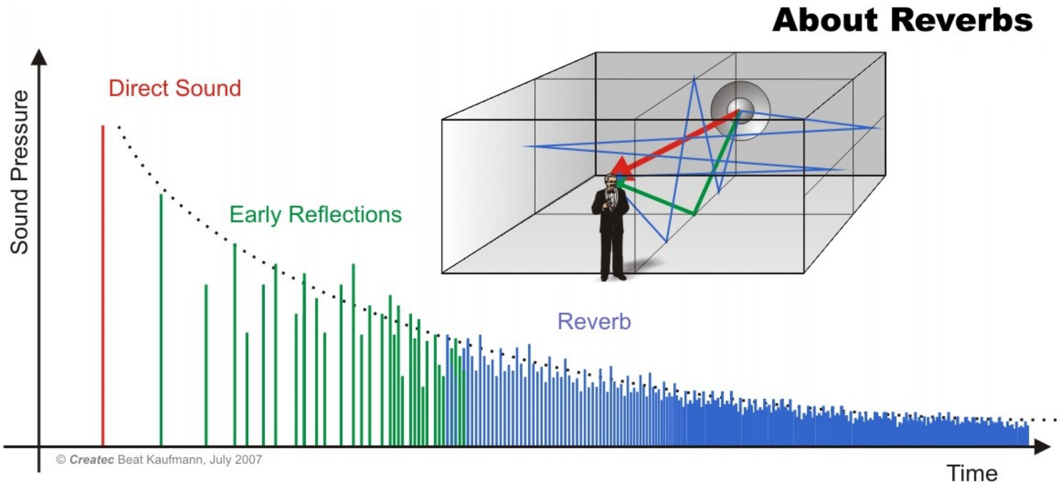

Figure 1.1: Example of a RIR showing the three phases of reverberation. The direct sound (red), the early reflections (green) and the late reverberation tail (blue). (Image source: [6])

1.2 Reverberation

Reverberation occurs in every scenario where sound is reflected during its propagation, thus it strongly depends on the surrounding environment. The received signal gets distorted on the acoustic pathway from the sound source to the receiver mainly by air absorption and reflections on any object in the environment. Hereby the latter is referred to as reverberation and responsible for the main distortion. Since reverberation is strongly correlated with the target speech it can be modeled well with a linear system filtering the original signal, which was emitted by any kind of acoustic source. These linear systems are often described by their impulse response, which is referred to as room impulse response (RIR) in scenarios inside of buildings, which are important environments in terms of sound communication applications. As shown in fig. 1.1 reverberation can usually be separated into three phases. The direct sound (red) corresponds to the sound wave propagating straight from the source to the receiver. The early reflections (green) correspond to sound waves being reflected only a few times so that their arrival times and directions are still distinct. They arrive up to approximately 50 ms later than the direct sound [7]. The late reverberation tail (blue) corresponds to the sound waves being reflected multiple times so that their arrival times and directions are not distinct anymore. Therefore the late reverberation is often described to be diffuse. Important to notice is that the early reflections are actually beneficial for speech intelligibility, since they can be integrated with the direct sound to increase the signal-to-noise ratio of the speech signal [8–11]. Therefore a need arises to develop algorithms, which attenuate the late reverber-

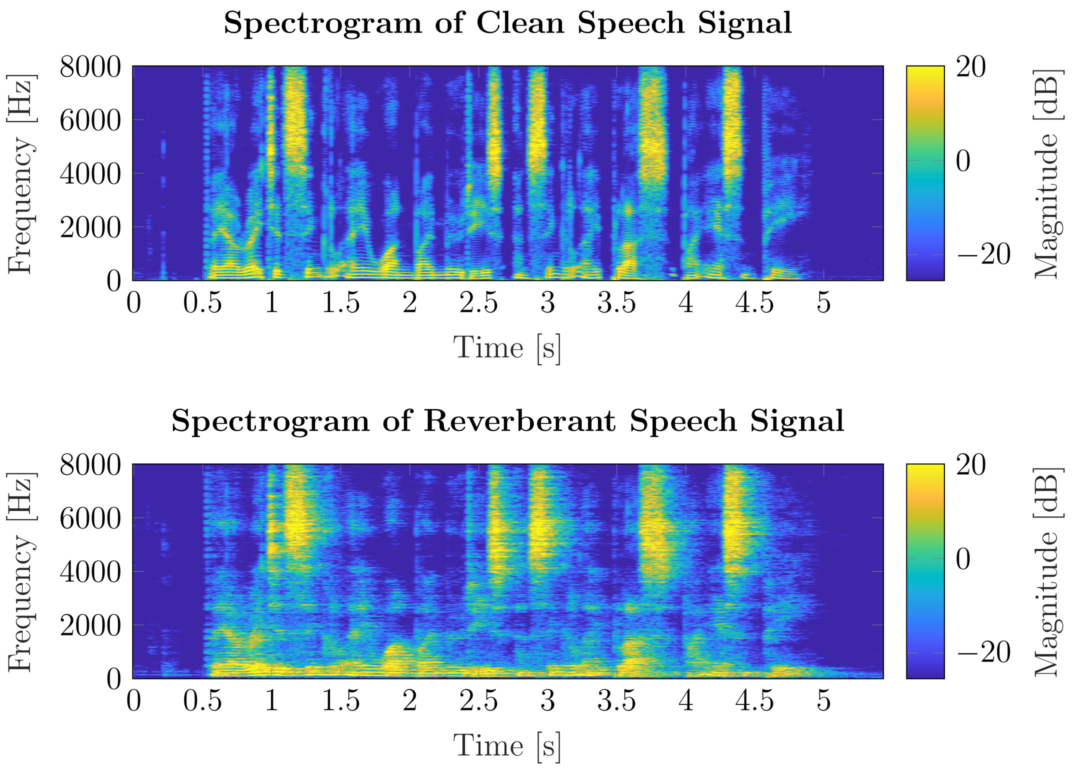

Figure 1.2: Spectrogram showing an example of the short time Fourier transform (STFT) coefficients of a clean speech signal (upper plot) and a corresponding reverberant speech signal (lower plot).

ation, but keep the early reflections. In order to develop such algorithms it is helpful to understand how the signal changes when reverberation is added to a clean speech signal. Comparing the spectra of the clean speech signal with the reverberant speech signal in fig. 1.2 reveals that the reverberation manipulates the speech signal to be less sparse in the time dimension. This is due to the fact that for each reflection additional energy is added to the direct signal with a certain time delay.

1.2.1 Dereverberation Algorithms

Here algorithms are introduced, which operate in the STFT domain on each frequency subband individually and which use the information (STFT-coefficients) of the past time frames to predict the reverberation in the present time frame, which then can be subtracted from the reverberant signal to extract the remaining desired signal. This method is referred to as multi-channel linear prediction (MCLP), since the multi-channel information of the past is used to predict the present reverberation by a linear filter [12, 13]. Hereby the signal is modeled with a convolutive acoustic transfer function (cATF) in the STFT domain and the prediction filter tries to invert the effect of the cATF. Additionally a sparsity promoting likelihood (LH) function could be utilized in order to optimize the prediction filter. Utilizing a time-varying complex Gaussian (TVG) model as sparse prior for the desired signal leads to the weighted prediction error (WPE) algorithm, which is widely used for dereverberation [14, 15]. Later a multiple-input multiple-output (MIMO) version of the WPE algorithm was proposed, which provides the possibility of further multi-channel speech enhancement methods (e.g. minimum power distortionless response (MPDR)-beamforming) [16]. In the next step the WPE algorithm was generalized using a complex generalized Gaussian (CGG) sparse prior and reformulated using the ℓ_p-norm of the desired signal as sparsity promoting cost function [17]. This reformulation was extended to a MIMO version utilizing the group sparsity structure in between the channels [18]. All of the mentioned algorithms perform dereverberation on batch signals, but the corresponding adaptive versions of these algorithms are also already proposed [19–21]. This thesis mainly focuses on the batch version of WPE, which only works for the stationary case, whereby the RIR is time-invariant within the batch duration, so that also the MCLP filter can be time-invariant.

1.3 Noise

Different kinds of noise are present in many everyday speech communication scenarios. Noise is usually modeled to be uncorrelated with the target speech component, which is reasonable, since the noise almost always comes from a different source. Therefore the uncorrelated noise is modeled to be additive to the clean or reverberant speech. Noise can be further categorized into spatially diffuse noise, i.e. the noise is not localizable, and spatially non-diffuse noise, e.g. an interfering speaker with a certain location. Another categorization is made by distinguishing stationary noise from non-stationary noise. Hereby the second order statistics (e.g. the covariance matrix) of stationary noise are constant over time, whereas they are time-varying for non-stationary noise. This thesis mainly focuses on diffuse stationary noise, which is modeled to be additive to the speech signal. Therefore only batch algorithms are described, which perform speech enhancement on a previously recorded signal batch. However all of the algorithms can also be formulated as an adaptive version, which is frame by frame online processing without looking into the future. Figure 1.3 shows the spectra containing the STFT coefficients of a clean speech signal (upper plot) and the corresponding

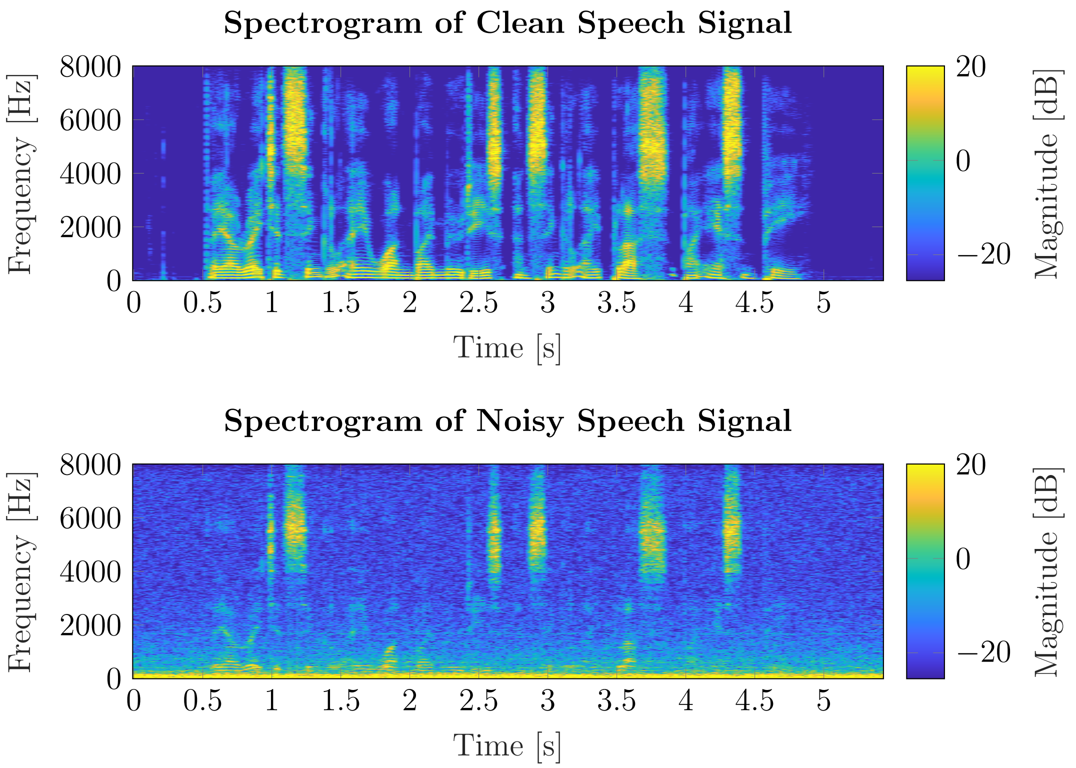

Figure 1.3: Spectrogram showing an example of the STFT coefficients of a clean speech signal (upper plot) and a corresponding speech signal with additive noise (lower plot).

noisy speech signal (lower plot), whereby diffuse stationary noise is added to the clean speech signal. The effect of the additive noise can be understood as blurring every information (STFT-coefficient) of the signal, which has less energy than the overshadowing noise.

1.3.1 Denoising Algorithms

The idea of common denoising algorithms is to filter a multi-channel signal with a linear filter, which is optimized utilizing an objective cost function [22]. Two widely known linear constrained beamformers are the MPDR beamformer, which is a linear filter minimizing the output power, and the minimum variance distortionless response (MVDR) beamformer, which is a linear filter minimizing the variance of the output corresponding to the noise component. However these two algorithms are equivalent in case a perfect relative transfer function (RTF) and perfect estimations of the noisy and noise covariance matrices are provided [23]. The estimation of an RTF is essential for MPDR and MVDR beamforming, since it introduces an additional constraint, which ensures a distortionless speech response, which means that the sound coming from the target speech direction should not be distorted at all. Of course some kind of knowledge of the target speech direction needs to be known. This knowledge end even more is usually given by an RTF of the target speech, which also needs to be estimated, so that it can be used in the described linear constrained beamformer. The delay-and-sum (DAS) structure is a basic arrangement of a beamformer. Its scheme is presented in fig. 1.4, where two sources are recorded of which only source one (red) is desired. This beamformer structure can extract this desired (red) signal, if the direction of source one is known, i.e. the corresponding delays $[\Delta_1, \Delta_2, \Delta_3, \Delta_4]$ and the corresponding weights $[w_1, w_2, w_3, w_4]$ are known. The combination of the delays and weights is equivalent to an multiplicative acoustic transfer function (mATF) in the STFT domain. The corresponding RTF is a normalized version of the mATF resulting from a division by the reference channel coefficient. One widely known and used algorithm to estimate the RTF vector is covariance whitening (CW) [24, 25]. Beamforming algorithms are usually multiple-input single-output (MISO) algorithms, which can also be seen in the DAS beamformer scheme in fig. 1.4.

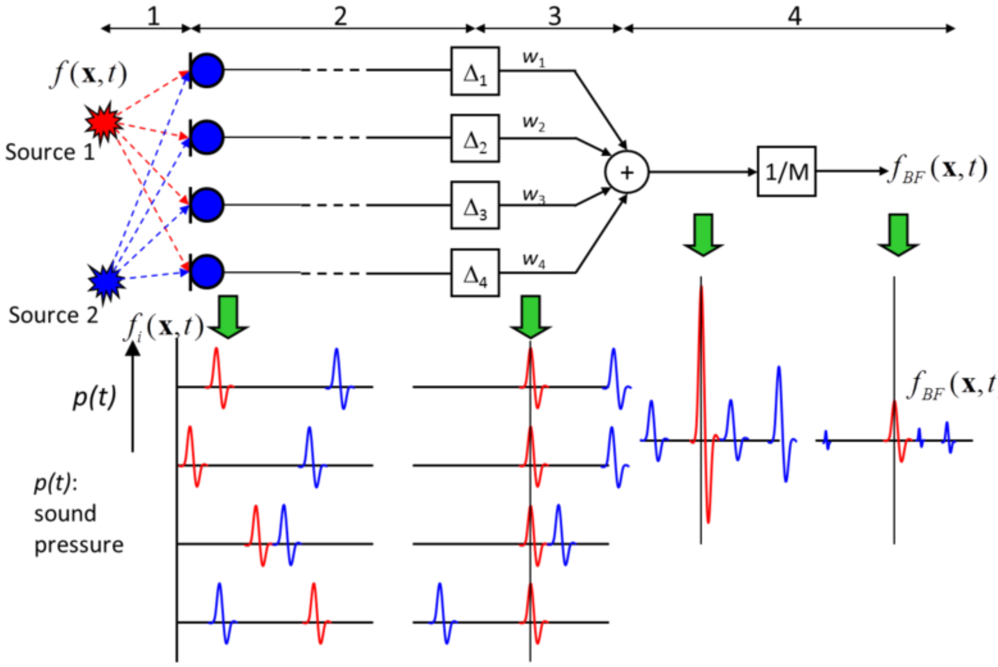

Figure 1.4: DAS structure with four input channels. Every channel is delayed and weighted before all channels are summed up. The delays and weights are chosen in a way to achieve constructive interference of the desired target source one (red). Additionally to the amplification of the target speech the undesired source (blue) is attenuated due to destructive interference. (Image source: [26])

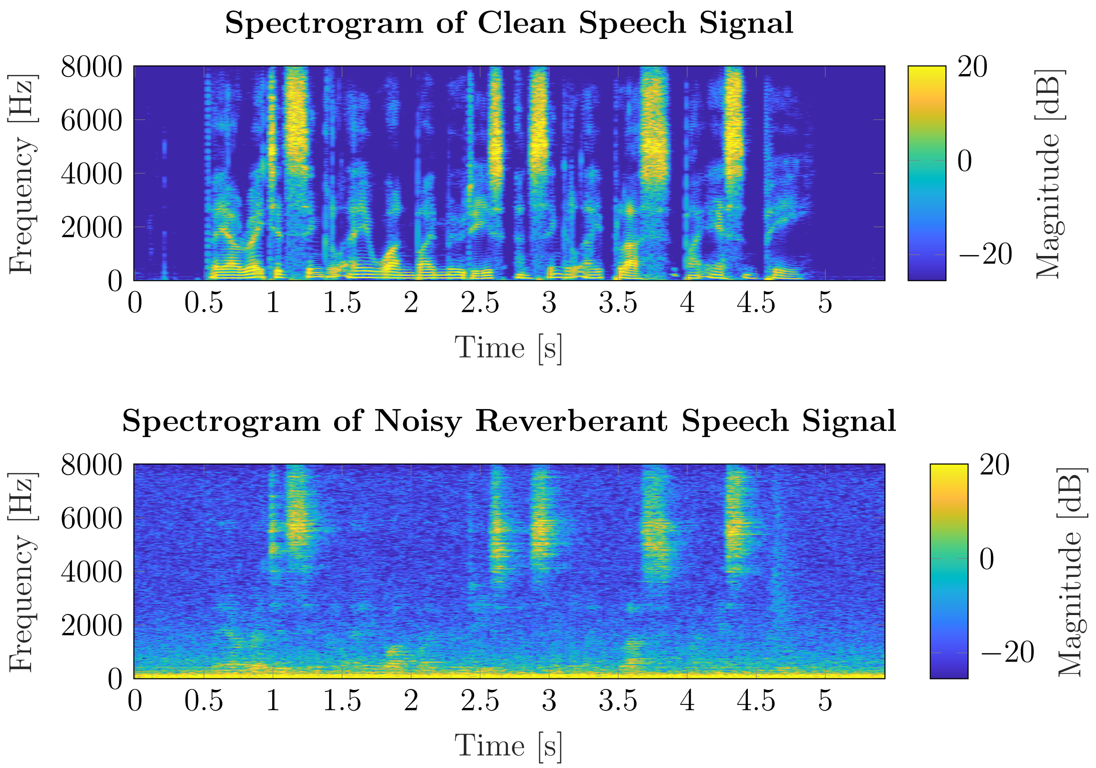

Figure 1.5: Spectrogram showing an example of the STFT coefficients of a clean speech signal (upper plot) and a corresponding speech signal with reverberation and additive noise (lower plot).

1.4 Reverberation and Noise

In many everyday situations both noise and reverberation deteriorates speech quality simultaneously. Figure 1.5 compares the spectra of such a noisy reverberant speech signal with its clean reference. From this representation it can be observed that the noisy reverberant signal is far less sparse than the clean speech, which is caused by both noise and reverberation. This would suggest to develop an algorithm performing unified dereverberation and denoising with a similar approach as for WPE.

1.4.1 Unified Algorithms

The MIMO extension of the conventional WPE algorithm provides the possibility of further multi-channel speech enhancement methods, such as linear constrained beamforming. Many cascade systems were proposed over the last years, whereby joint dereverberation and denoising is performed in two stages, e.g. as in [27, 28]. Often the first stage is the MIMO-WPE algorithm and the second stage e.g. an MPDR algorithm. Hereby the optimization of the two different filters happens separately. However, recently a novel algorithm was proposed by Nakatani et al., which unifies these two stages so that it performs a joint optimization of the MIMO-WPE filter and the MPDR beamformer filter. This algorithm is referred to as weighted power minimization distortionless response (WPD) [1, 29, 30] and lays the foundation of this thesis, which aims at extending and improving this WPD algorithm with a new cost function and a MIMO version. Another recently proposed approach utilizes a generalized sidelobe canceler (GSC) structure for joint dereverberation and noise reduction [31–33].

Chapter 2

Theory

This chapter describes the underlying signal model for convolutional beamformers and some of the conventional theory of dereverberation and denoising algorithms. Although all of the following derivations can be done similarly in the time domain, everything throughout this thesis is formulated in the STFT domain, whereby the frequency subbands (denoted by k) are processed individually in a parallel manner. The general signal model is described in section 2.1. Hereafter the derivations of the following algorithms are described:

- MPDR/MVDR denoising algorithm (section 2.2)
- CW estimating an RTF (section 2.3)
- WPE algorithm (section 2.4)
  - MISO-WPE with TVG model (section 2.4.2.1)
  - MISO-WPE with general sparse prior (section 2.4.2.2)
  - MISO-WPE with CGG sparse prior (section 2.4.2.3)
  - MISO-WPE with ℓ_p-norm cost function (section 2.4.2.4)
  - MIMO-WPE with ℓ_p-norm cost function (section 2.4.2.5)
- WPD algorithm with TVG model (section 2.5)

2.1 Signal Model

The origin of the signal to be enhanced using convolutional beamforming is the clean speech signal $s_{k,t}$ uttered by a speaker. Here the clean speech is modeled in the STFT domain (narrow band signal model), whereby the indices k and t denote the frequency bin and time frame respectively. Then the signal is captured by M microphones in a noisy reverberant environment resulting in the noisy signal

$$\boxed{\mathbf{y}_{k,t} = \underbrace{\mathbf{d}_{k,t} + \mathbf{r}_{k,t}}_{\mathbf{x}_{k,t}} + \mathbf{n}_{k,t} \quad \forall k,t} \tag{2.1}$$

where $\mathbf{y}_{k,t} = \left[y_{k,t}^{(1)}, y_{k,t}^{(2)}, \dots, y_{k,t}^{(M)}\right]^T \in \mathbb{C}^M$ contains the STFT coefficients of the multi-channel microphone signal with $\bullet^T$ denoting the transpose of $\bullet$. The stacked signal vectors $\mathbf{x}_{k,t}$, $\mathbf{d}_{k,t}$, $\mathbf{r}_{k,t}$ and $\mathbf{n}_{k,t}$ similarly contain the STFT coefficients of the multi-channel reverberant speech, the direct path and early reflections, the late reverberation tail and the additive noise respectively. Using the clean speech $s_{k,t}$ and a stacked filter vector $\mathbf{a}_{k,l}$ containing the STFT coefficients of the multi-channel cATF can further describe the reverberant speech. It is to be noticed that the cATF is an STFT domain approximation of the multi-channel RIR in the time domain. The first part with the desired signal is then given by

$$\mathbf{d}_{k,t} = \sum_{l=0}^{\tau_k - 1} \mathbf{a}_{k,l} s_{k,t-l} \approx \mathbf{v}_k s_{k,t} \approx \tilde{\mathbf{v}}_k^{(m)} d_{k,t}^{(m)} \quad \forall k,t \tag{2.2}$$

where τ_k is the prediction delay separating the early reflections from the late reverberation tail, l is the filter tap index and L_k the frequency dependent filter length. The second part with the reverberation tail is then given by

$$\mathbf{r}_{k,t} = \sum_{l=\tau_k}^{L_k - 1} \mathbf{a}_{k,l} s_{k,t-l} \quad \forall k,t \tag{2.3}$$

In eq. (2.2) the desired signal $\mathbf{d}_{k,t}$ is approximated multiplying the mATF $\mathbf{v}_k$ with the clean speech $s_{k,t}$, which is feasible under the assumption that the analysis window of the STFT framework is longer than the duration of the early reflections in the time domain. The RTF is defined as $\tilde{\mathbf{v}}_k^{(m)} = \mathbf{v}_k / v_k^{(m)}$, where m corresponds to the reference channel so that $d_{k,t}^{(m)} = v_k^{(m)} s_{k,t}$ holds.

The goal of convolutional beamforming is to subtract the reverberation tail $\mathbf{r}_{k,t}$ and additive noise $\mathbf{n}_{k,t}$ from the noisy microphone signal $\mathbf{y}_{k,t}$ to extract the desired signal $\mathbf{d}_{k,t}$ containing the direct speech and early reflections.

2.2 MPDR and MVDR Beamforming

MPDR and MVDR beamforming both use a linear filter to extract the target signal from a certain direction out of the multi-channel microphone signals. For this both algorithms suppress the noise component, which is assumed to be diffuse. In order to find the optimal filter a cost function is set up so that either the noisy signal power (MPDR) or the power of the noise component (MVDR) should be minimized. In order to maintain the desired target speech component at the output of the beamformer a linear constraint is enforced on the optimal filter, so that every signal corresponding to the target speech RTF will not be distorted. The following derivations are mainly based on [22, 23].

2.2.1 Signal Model and Beamformer

The signal model is similar to eq. (2.1), but for now a free-field assumption is made, whereby no reverberation is present in the signal, so that the recorded microphone signal $\mathbf{y}_{k,t}$ only consist out of the desired speech signal $\mathbf{d}_{k,t}$ and the additive noise $\mathbf{n}_{k,t}$.

$$\mathbf{y}_{k,t} = \underbrace{\mathbf{d}_{k,t}}_{\mathbf{x}_{k,t}} + \mathbf{n}_{k,t}, \tag{2.4}$$

The goal of MPDR and MVDR beamforming is to extract the single-channel target speech component $d_{k,t}^{(m)}$, which corresponds to the single-channel clean speech $s_{k,t}$, from the noisy signal $\mathbf{y}_{k,t}$ by applying the filter $\mathbf{q}_k^{(m)}$. The beamformed signal is then given as

$$z_{k,t}^{(m)} = \mathbf{q}_k^{(m)H} \mathbf{y}_{k,t} \tag{2.5}$$

whereby $\bullet^H$ denotes the conjugate-transpose (hermitian) of a vector or matrix.

2.2.2 Filter optimization

In order to optimize the filter coefficients $\mathbf{q}_k^{(m)}$ a cost function is set up, which corresponds to the objective. In the case of the MPDR beamformer the power of the complete noisy signal $\mathbf{y}_{k,t}$ should be minimized, but additionally the target speech should not be distorted, which can be enforced by constraining the optimization using the target speech RTF. Formally we have

$$\begin{aligned}\mathbf{q}_k^{(m)\text{MPDR}} &= \underset{\mathbf{q}_k^{(m)}}{\operatorname{argmin}}\, \mathbb{E}\left[\left|z_{k,t}^{(m)}\right|^2\right] \quad \text{s.t.} \quad \mathbf{q}_k^{(m)H}\tilde{\mathbf{v}}_k^{(m)} \overset{!}{=} 1 \\ &= \underset{\mathbf{q}_k^{(m)}}{\operatorname{argmin}}\, \mathbf{q}_k^{(m)H}\mathbb{E}\left[\mathbf{y}_{k,t}\mathbf{y}_{k,t}^H\right]\mathbf{q}_k^{(m)} \quad \text{s.t.} \quad \mathbf{q}_k^{(m)H}\tilde{\mathbf{v}}_k^{(m)} \overset{!}{=} 1 \\ &= \underset{\mathbf{q}_k^{(m)}}{\operatorname{argmin}}\, \mathbf{q}_k^{(m)H}\mathbf{R}_{y,k}\mathbf{q}_k^{(m)} \quad \text{s.t.} \quad \mathbf{q}_k^{(m)H}\tilde{\mathbf{v}}_k^{(m)} \overset{!}{=} 1\end{aligned} \tag{2.6}$$

whereby $\mathbb{E}\left[\mathbf{y}_{k,t}\mathbf{y}_{k,t}^H\right]$ is the definition of the noisy covariance matrix $\mathbf{R}_{y,k}$ with $\mathbb{E}\left[\bullet\right]$ being the expectation value operator. In the case of MVDR beamforming the power of the noise signal $\mathbf{n}_{k,t}$ should be minimized using the same constraint as in MPDR beamforming. It follows

$$\begin{aligned}\mathbf{q}_k^{(m)\text{MVDR}} &= \underset{\mathbf{q}_k^{(m)}}{\operatorname{argmin}}\, \mathbf{q}_k^{(m)H}\mathbb{E}\left[\mathbf{n}_{k,t}\mathbf{n}_{k,t}^H\right]\mathbf{q}_k^{(m)} \quad \text{s.t.} \quad \mathbf{q}_k^{(m)H}\tilde{\mathbf{v}}_k^{(m)} \overset{!}{=} 1 \\ &= \underset{\mathbf{q}_k^{(m)}}{\operatorname{argmin}}\, \mathbf{q}_k^{(m)H}\mathbf{R}_{n,k}\mathbf{q}_k^{(m)} \quad \text{s.t.} \quad \mathbf{q}_k^{(m)H}\tilde{\mathbf{v}}_k^{(m)} \overset{!}{=} 1\end{aligned} \tag{2.7}$$

The solutions to these optimization problems are widely know as

$$\mathbf{q}_k^{(m)\text{MPDR}} = \frac{\mathbf{R}_{y,k}^{-1}\tilde{\mathbf{v}}_k^{(m)}}{\tilde{\mathbf{v}}_k^{(m)H}\mathbf{R}_{y,k}^{-1}\tilde{\mathbf{v}}_k^{(m)}} \tag{2.8}$$

$$\mathbf{q}_k^{(m)\text{MVDR}} = \frac{\mathbf{R}_{n,k}^{-1}\tilde{\mathbf{v}}_k^{(m)}}{\tilde{\mathbf{v}}_k^{(m)H}\mathbf{R}_{n,k}^{-1}\tilde{\mathbf{v}}_k^{(m)}} \tag{2.9}$$

In order to determine the optimal filter $\mathbf{q}_k^{(m)}$ estimations of the noisy and noise covariance matrices and the target speech RTF vector are necessary. The latter can be estimated using CW described in section 2.3.

2.2.3 Estimation of Covariance Matrices

There are two scenarios for the estimation of the covariance matrices and RTF. For real time applications online signal processing is necessary, whereby the filter coefficients are adaptive over time. Hereby also the covariance matrices and RTF need to be estimated adaptively, e.g. with recursive smoothing. If the task is signal enhancement of already recorded data and the signal statistics can be assumed to be stationary, it is possible to perform batch signal processing, whereby a whole batch of the signal (can be multiple seconds) is used for the

Algorithm 1: MPDR/MVDR (batch)

```
input      : batch-matrix of multi-channel noisy microphone signal $\mathbf{Y}_k \; \forall \; k$
parameters: reference channel $m$, beamformer $b \in \{\text{"MPDR"}, \text{"MVDR"}\}$
functions  : speech presence probability $\texttt{spp}\,(\bullet)$, trace of a matrix $\texttt{trace}\,(\bullet)$,
             covariance whitening $\texttt{cw}\,(\bullet)$
output     : batch-vector of single-channel beamformed signal $\mathbf{z}_k^{(m)} \; \forall \; k$

foreach $k \in \{1, 2, \dots, K\}$ do                 // process each frequency subband $k$ individually
    $\mathbf{R}_{y,k} = \frac{1}{T}\mathbf{Y}_k\mathbf{Y}_k^H$                 // estimating noisy covariance matrix by SCM
    $\mathbf{\Sigma}_k = \texttt{spp}\,(\mathbf{Y}_k)$                 // estimating SPP of the noisy signal
    $\mathbf{R}_{n,k} = \frac{\mathbf{Y}_k(\mathbf{I}_{\mathbb{N}} - \mathbf{\Sigma}_k)\mathbf{Y}_k^H}{\texttt{trace}(\mathbf{I}_{\mathbb{N}} - \mathbf{\Sigma}_k)}$     // estimating noise covariance matrix by SCM using the SPP
    $\tilde{\mathbf{v}}_k^{(m)} = \texttt{cw}\,(\mathbf{Y}_k, \mathbf{R}_{n,k}, m)$     // estimating RTF-vector by CW (algorithm 2)
    switch $b$ do                                     // select beamformer
        case "MPDR" do
            $\mathbf{q}_k^{(m)} = \frac{\mathbf{R}_{y,k}^{-1}\tilde{\mathbf{v}}_k^{(m)}}{\tilde{\mathbf{v}}_k^{(m)H}\mathbf{R}_{y,k}^{-1}\tilde{\mathbf{v}}_k^{(m)}}$     // optimal MPDR beamformer
        case "MVDR" do
            $\mathbf{q}_k^{(m)} = \frac{\mathbf{R}_{n,k}^{-1}\tilde{\mathbf{v}}_k^{(m)}}{\tilde{\mathbf{v}}_k^{(m)H}\mathbf{R}_{n,k}^{-1}\tilde{\mathbf{v}}_k^{(m)}}$     // optimal MVDR beamformer
    $\mathbf{z}_k^{(m)} = \mathbf{q}_k^{(m)H}\mathbf{Y}_k$                 // beamforming the noisy signal
```

estimations of the covariance matrices and RTF. The noisy covariance matrix in the batch case can be estimated using its sample covariance matrix (SCM) as

$$\mathbf{R}_{y,k} = \mathbb{E}\left[\mathbf{y}_{k,t}\mathbf{y}_{k,t}^H\right] \approx \frac{1}{T}\mathbf{Y}_k\mathbf{Y}_k^H = \frac{1}{T}\sum_{t=1}^{T}\mathbf{y}_{k,t}\mathbf{y}_{k,t}^H \in \mathbb{C}^{M\times M} \tag{2.10}$$

whereby T is the number of time frames and $\mathbf{Y}_k = [\mathbf{y}_{k,1}, \mathbf{y}_{k,2}, \dots, \mathbf{y}_{k,T}] \in \mathbb{C}^{M\times T}$ is the batch matrix of the noisy signal. The noise covariance matrix can also be estimated using its SCM as

$$\mathbf{R}_{n,k} = \mathbb{E}\left[\mathbf{n}_{k,t}\mathbf{n}_{k,t}^H\right] \approx \frac{1}{T}\sum_{t=1}^{T}\mathbf{n}_{k,t}\mathbf{n}_{k,t}^H \in \mathbb{C}^{M\times M} \tag{2.11}$$

but since the additive noise signal is not known in the blind case, it can be approximated by applying an inverse of the speech presence probability (SPP) $\sigma_{k,t}$ on the noisy microphone signal $\mathbf{y}_{k,t}$ like this:

$$\mathbf{R}_{n,k} \approx \frac{\sum_{t=1}^{T}\left(1 - \sigma_{k,t}\right)\mathbf{y}_{k,t}\mathbf{y}_{k,t}^H}{\sum_{t=1}^{T}\left(1 - \sigma_{k,t}\right)} \tag{2.12}$$

The SPP gives an estimate of the probability that speech is present for the k^{th}

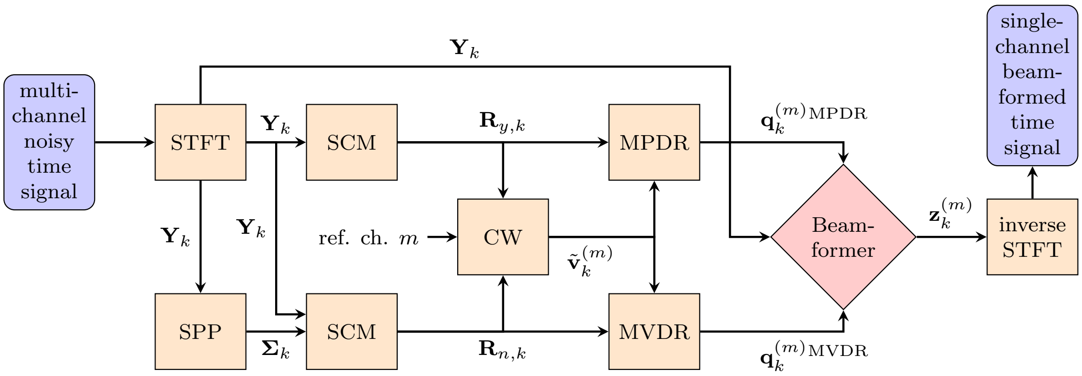

Figure 2.1: Flow chart of MPDR/MVDR beamforming in the STFT domain.

frequency subband and the t^{th} time frame and it can be estimated e.g. as in [34]. The inverse probability is given by subtracting the SPP from one and it indicates a noise-only-presence probability. All frames containing only noise can be used for estimation of the noise covariance matrix. An overview of the complete workflow of the conventional MPDR/MVDR-beamforming algorithm is presented in fig. 2.1 and algorithm 1.

2.3 Covariance whitening (CW)

The aforementioned MPDR and MVDR beamformers are derived with a constraint involving the target speech RTF. CW is a widely used method to estimate this target speech RTF. The following derivations are mainly based on [24, 25]. Here the signal model already described in eq. (2.4) is utilized and rewritten in terms of covariance matrices, which can be done assuming that the noise signal $\mathbf{n}_{k,t}$ is uncorrelated with the desired speech signal $\mathbf{d}_{k,t}$.

$$\mathbf{R}_{y,k} = \mathbf{R}_{d,k} + \mathbf{R}_{n,k} \tag{2.13}$$

Using the approximation with an mATF in eq. (2.2) allows to decompose the signal model as

$$\mathbf{R}_{y,k} = \underbrace{\phi_{s,k}\mathbf{v}_k\mathbf{v}_k^H}_{\mathbf{R}_{d,k}} + \mathbf{R}_{n,k} \tag{2.14}$$

where $\phi_{s,k} = \mathbb{E}\left[s_{k,t}s_{k,t}^*\right] \approx \frac{1}{T}\sum_{t=1}^{T}|s_{k,t}|^2 \in \mathbb{R}_{\geq 0}$ is the speech power spectral density (PSD). Hereby it can be observed that the desired speech covariance matrix has a rank of one, since it is a scaled version of the multiplication of the mATF with its hermitian. Therefore an eigenvalue decomposition (EVD) of the speech covariance matrix as in principal component analysis (PCA) could be used to estimate the mATF, whereby the eigenvector corresponding to the largest eigenvalue indicates the principal component. This so called maximal eigenvector is an arbitrary scaled version of the mATF, whereby its normalized version is the RTF we are looking for. However this only holds if the target speech has the highest power of all spatially non-diffuse sources in the signal, which is true for the condition of one localized target speech source mixed with additive diffuse noise. The speech covariance matrix can also be estimated using its SCM and looks like this:

$$\mathbf{R}_{d,k} = \mathbb{E}\left[\mathbf{d}_{k,t}\mathbf{d}_{k,t}^H\right] \approx \frac{1}{T}\sum_{t=1}^{T}\mathbf{d}_{k,t}\mathbf{d}_{k,t}^H \in \mathbb{C}^{M\times M} \tag{2.15}$$

However since the speech component is not available in the blind case an approximation method [24, 25] can be used, which performs pre-whitening of the noisy signal. The pre-whitened signal $\mathbf{w}_{k,t}$ is given by multiplying the inverse square root of the noise covariance matrix $\mathbf{R}_{n,k}$ with the noisy covariance matrix $\mathbf{R}_{y,k}$ like this:

$$\mathbf{w}_{k,t} = \mathbf{R}_{n,k}^{-H/2}\mathbf{y}_{k,t} \tag{2.16}$$

Hereby the noise covariance matrix can be estimated as described in section 2.2.3 and a square root of it can be defined by

$$\mathbf{R}_{n,k} = \mathbf{R}_{n,k}^{H/2}\mathbf{R}_{n,k}^{1/2} \quad \Rightarrow \quad \sqrt{\mathbf{R}_{n,k}} = \mathbf{R}_{n,k}^{1/2} \quad \text{and} \quad \mathbf{R}_{n,k}^{H/2} = \left(\mathbf{R}_{n,k}^{1/2}\right)^H \tag{2.17}$$

since the covariance matrix is a hermitian and positive definite matrix. It is to be noticed that the square root is not unique, but for this derivation it does not matter, whether the square root is determined using the EVD or the Cholesky square root or any other method, as long as eq. (2.17) is fulfilled. The pre-whitened covariance matrix $\mathbf{R}_{w,k}$ can be determined using its SCM like this:

$$\begin{aligned}\mathbf{R}_{w,k} &= \mathbb{E}\left[\mathbf{w}_{k,t}\mathbf{w}_{k,t}^H\right] \approx \frac{1}{T}\sum_{t=1}^{T}\mathbf{w}_{k,t}\mathbf{w}_{k,t}^H \in \mathbb{C}^{M\times M}\\ &= \mathbf{R}_{n,k}^{-H/2}\mathbf{R}_{y,k}\mathbf{R}_{n,k}^{-1/2}\\ &= \mathbf{R}_{n,k}^{-H/2}\mathbf{R}_{d,k}\mathbf{R}_{n,k}^{-1/2} + \underbrace{\mathbf{R}_{n,k}^{-H/2}\mathbf{R}_{n,k}\mathbf{R}_{n,k}^{-1/2}}_{\mathbf{I}_M}\\ &= \phi_{s,k}\mathbf{R}_{n,k}^{-H/2}\mathbf{v}_k\mathbf{v}_k^H\mathbf{R}_{n,k}^{-1/2} + \mathbf{I}_M\end{aligned} \tag{2.18}$$

Now the principal component of this pre-whitened covariance matrix $\mathbf{R}_{w,k}$ is extracted using its EVD, which is defined as $\mathbf{R}_{w,k} = \mathbf{\Psi}_{w,k}\mathbf{\Gamma}_{w,k}\mathbf{\Psi}_{w,k}^H$, since this

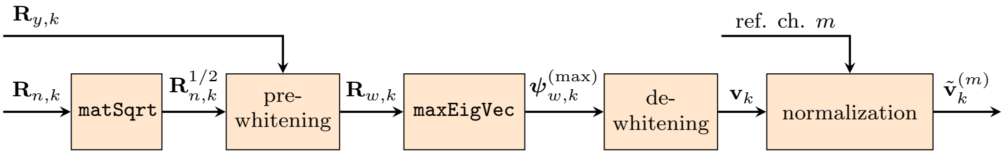

Figure 2.2: Flow chart of CW.

pre-whitened covariance matrix is hermitian. Hereby $\boldsymbol{\Gamma}_{w,k}$ is a diagonal matrix consisting of the sorted eigenvalues of $\mathbf{R}_{w,k}$ and $\boldsymbol{\Psi}_{w,k}$ is the corresponding eigenvector matrix. The maximal eigenvalue is also the first eigenvalue in $\boldsymbol{\Gamma}_{w,k}$ so that the eigenvector $\boldsymbol{\psi}_{w,k}^{(\max)}$ corresponding to the principal component can be extracted as the first column of $\boldsymbol{\Psi}_{w,k}$. In order to get an arbitrarily scaled estimation of the target speech mATF a de-whitening operation using the square root of the noise covariance matrix is performed as

$$\mathbf{v}_k = \mathbf{R}_{n,k}^{H/2} \boldsymbol{\psi}_{w,k}^{(\max)} \tag{2.19}$$

The RTF can now be deduced from the mATF $\mathbf{v}_k = \left[v_k^{(1)}, v_k^{(2)}, \ldots, v_k^{(M)} \right]^T \in \mathbb{C}^M$ by simply dividing by the entry of the reference channel m like this:

$$\tilde{\mathbf{v}}_k^{(m)} = \frac{\mathbf{v}_k}{v_k^{(m)}} = \frac{\mathbf{R}_{n,k}^{H/2} \boldsymbol{\psi}_{w,k}^{(\max)}}{\mathbf{e}_M^{(m)T} \mathbf{R}_{n,k}^{H/2} \boldsymbol{\psi}_{w,k}^{(\max)}} \tag{2.20}$$

An overview of CW is summarized by fig. 2.2 and algorithm 2.

Algorithm 2: Covariance Whitening (batch)

input : batch-matrix of multi-channel noisy microphone signal $\mathbf{Y}_k$, multi-channel noise covariance matrix $\mathbf{R}_{n,k}$

parameters: reference channel m

functions : matrix square-root `matSqrt` (•), eigenvector corresponding to maximal eigenvalue `maxEigVec` (•)

output : RTF-vector $\tilde{\mathbf{v}}_k^{(m)}$ for one frequency subband k corresponding to reference channel m

```
$\mathbf{R}_{n,k}^{1/2} = \texttt{matSqrt}(\mathbf{R}_{n,k})$    // calculating arbitrarily matrix square root
$\mathbf{W}_k = \mathbf{R}_{n,k}^{-H/2} \mathbf{Y}_k$    // pre-whitening of noisy signal
$\mathbf{R}_{w,k} = \frac{1}{T} \mathbf{W}_k \mathbf{W}_k^H$    // whitened covariance matrix (estimated by SCM)
$\boldsymbol{\psi}_{w,k}^{(\max)} = \texttt{maxEigVec}(\mathbf{R}_{w,k})$    // extracting principal component
$\mathbf{v}_k = \mathbf{R}_{n,k}^{H/2} \boldsymbol{\psi}_{w,k}^{(\max)}$    // de-whitening of principal component
$\tilde{\mathbf{v}}_k^{(m)} = \mathbf{v}_k / v_k^{(m)}$    // calculating RTF by normalization with reference channel $m$
```

2.4 WPE Dereverberation

Since reverberation is caused by reflections it is a delayed and distorted version of the direct sound from past frames as described in section 1.2. Therefore the idea of WPE dereverberation is to predict the late reverberation $\mathbf{r}_{k,t}$ of the current time frame t by a linear prediction filter $\tilde{\mathbf{g}}_k^{(m)}$ applied to the past frames of the noisy multi-channel microphone signal $\mathbf{y}_{k,t}$. The method of such an algorithm is also referred to as MCLP [12, 14, 15]. The so called prediction delay τ_k determines the size of the gap between the current time frame and the past time frames, which correspond to the late reverberation $\mathbf{r}_{k,t}$. Its importance lies in keeping the early reflections, which are beneficial for speech intelligibility as described in section 1.2, and in avoiding whitening of the filtered signal $\breve{\mathbf{d}}_{k,t}$. The filter optimization takes advantages of the difference in sparsity of the reverberant and clean speech signal. Conventionally a TVG is used as sparsity promoting LH function $\mathbb{L}$ to model the single-channel desired signal $d_{k,t}^{(m)}$ (see section 2.4.2.1), however it can be reformulated and generalized using the ℓ_p-norm (section 2.4.2.4), which can be further extended to perform MIMO-WPE (section 2.4.2.5).

2.4.1 Signal Model and MCLP Filter

The signal model is similar to eq. (2.1), but for now a noise-free assumption is made, whereby the recorded microphone signal $\mathbf{y}_{k,t}$ only consists of the desired speech signal $\mathbf{d}_{k,t}$ and the late reverberation tail $\mathbf{r}_{k,t}$.

$$\mathbf{y}_{k,t} = \underbrace{\mathbf{d}_{k,t} + \mathbf{r}_{k,t}}_{\mathbf{x}_{k,t}} \tag{2.21}$$

The goal of WPE dereverberation is now to estimate the late reverberation $r_{k,t}^{(m)}$ using the prediction filter $\tilde{\mathbf{g}}_k^{(m)}$ and subtracting it from the noisy microphone signal $y_{k,t}^{(m)}$ of the reference channel m in order to extract the single-channel dereverberated speech signal $\breve{d}_{k,t}^{(m)}$.

$$\breve{d}_{k,t}^{(m)} = y_{k,t}^{(m)} - r_{k,t}^{(m)} = y_{k,t}^{(m)} - \underbrace{\tilde{\mathbf{g}}_k^{(m)H}\tilde{\mathbf{y}}_{k,t}}_{r_{k,t}^{(m)}} = \bar{\mathbf{g}}_k^{(m)H}\bar{\mathbf{y}}_{k,t} \tag{2.22}$$

whereby the reformulated prediction filter $\bar{\mathbf{g}}_k^{(m)}$ is defined as

$$\bar{\mathbf{g}}_k^{(m)} = \begin{bmatrix} \mathbf{e}_M^{(m)} \\ -\tilde{\mathbf{g}}_k^{(m)} \end{bmatrix} \in \mathbb{C}^{M(L_k - \tau_k + 1)} \tag{2.23}$$

Hereby $\tilde{\mathbf{y}}_{k,t} = \left[\mathbf{y}_{k,t-\tau_k}^T, \mathbf{y}_{k,t-\tau_k-1}^T, \mathbf{y}_{k,t-\tau_k-2}^T, \ldots, \mathbf{y}_{k,t-L_k+1}^T\right]^T \in \mathbb{C}^{M(L_k-\tau_k)}$ is the stacked signal vector containing the past frames corresponding to the late reverberation $\mathbf{r}_{k,t}$ and $\bar{\mathbf{y}}_{k,t} = \left[\mathbf{y}_{k,t}^T, \tilde{\mathbf{y}}_{k,t}^T\right]^T \in \mathbb{C}^{M(L_k-\tau_k+1)}$ is the combined stacked signal vector with the momentary frame $\mathbf{y}_{k,t}$ and the past frames $\tilde{\mathbf{y}}_{k,t}$. Notice that there is a gap of $\tau_k - 1$ time frames between the first and second entry of $\bar{\mathbf{y}}_{k,t}$. This can also be formulated for the whole batch as

$$\breve{\mathbf{d}}_k^{(m)} = \mathbf{y}_k^{(m)} - \tilde{\mathbf{g}}_k^{(m)H} \tilde{\mathbf{Y}}_k = \bar{\mathbf{g}}_k^{(m)H} \bar{\mathbf{Y}}_k \tag{2.24}$$

with $\breve{\mathbf{d}}_k^{(m)} = \left[\breve{d}_{k,1}^{(m)}, \breve{d}_{k,2}^{(m)}, \ldots, \breve{d}_{k,T}^{(m)}\right] \in \mathbb{C}^{1\times T}$, $\mathbf{y}_k^{(m)} = \left[y_{k,1}^{(m)}, y_{k,2}^{(m)}, \ldots, y_{k,T}^{(m)}\right] \in \mathbb{C}^{1\times T}$, $\tilde{\mathbf{Y}}_k = [\tilde{\mathbf{y}}_{k,1}, \tilde{\mathbf{y}}_{k,2}, \ldots, \tilde{\mathbf{y}}_{k,T}] \in \mathbb{C}^{M(L_k-\tau_k)\times T}$ and $\bar{\mathbf{Y}}_k = [\bar{\mathbf{y}}_{k,1}, \bar{\mathbf{y}}_{k,2}, \ldots, \bar{\mathbf{y}}_{k,T}] \in \mathbb{C}^{M(L_k-\tau_k+1)\times T}$ being the stacked batch vectors of $\breve{d}_{k,t}^{(m)}$ and $y_{k,t}^{(m)}$ and the stacked batch matrices of $\tilde{\mathbf{y}}_{k,t}$ and $\bar{\mathbf{y}}_{k,t}$ respectively.

2.4.2 Filter Optimization

In order to find the optimal filter a LH function is set up according to the assumption that a dereverberated signal is more sparse than its corresponding original. The following subsections show five similar approaches to optimize the WPE dereverberation filter.

2.4.2.1 Conventional MCLP Dereverberation using a TVG Model

The conventional WPE algorithm is derived using the TVG model as sparse prior for the single-channel desired signal $d_{k,t}^{(m)}$ [17], corresponding to the reference channel m. Hereby the distribution ρ of the STFT coefficients is modeled by a circular complex Gaussian probability density function (PDF) $\mathcal{N}_{\mathbb{C}}$

$$\rho\left(d_{k,t}^{(m)}, \lambda_{k,t}\right) = \mathcal{N}_{\mathbb{C}}\left(d_{k,t}^{(m)}; 0, \lambda_{k,t}\right) = \frac{1}{\pi\lambda_{k,t}} e^{-\frac{\left|d_{k,t}^{(m)}\right|^2}{\lambda_{k,t}}} \tag{2.25}$$

with a zero mean and an unknown and time-varying variance $\lambda_{k,t} \in \mathbb{R}_{>0}$, which corresponds to the power of the desired speech signal. The LH as product of the probabilities of all T time frames is then given by

$$\mathbb{L}\left(\mathbf{d}_k^{(m)}, \boldsymbol{\lambda}_k\right) = \prod_{t=1}^{T} \mathcal{N}_{\mathbb{C}}\left(d_{k,t}^{(m)}; 0, \lambda_{k,t}\right) \tag{2.26}$$

whereby $\mathbf{d}_k^{(m)} = \left[d_{k,1}^{(m)}, d_{k,2}^{(m)}, \ldots, d_{k,T}^{(m)}\right] \in \mathbb{C}^{1\times T}$ and $\boldsymbol{\lambda}_k = [\lambda_{k,1}, \lambda_{k,2}, \ldots, \lambda_{k,T}] \in \mathbb{R}_{>0}^{1\times T}$ are the batch vectors containing the desired signal and the variances of the reference channel m for each time frame t respectively. The aim is to find the optimal MCLP filter $\tilde{\mathbf{g}}_k^{(m)}$ and the corresponding variances $\boldsymbol{\lambda}_k$ which maximize the LH $\mathbb{L}$, i.e. the solution to

$$\tilde{\mathbf{g}}_k^{(m)\mathrm{WPE}} = \underset{\tilde{\mathbf{g}}_k^{(m)},\boldsymbol{\lambda}_k>0}{\operatorname{argmax}} \mathbb{L}\left(\mathbf{d}_k^{(m)}, \boldsymbol{\lambda}_k\right) \tag{2.27}$$

Alternatively to maximizing the LH $\mathbb{L}$ the negative log-likelihood (nLLH) $\mathcal{L}$ can be seen as a cost function that needs to be minimized to find the optimal prediction filter

$$\begin{aligned}\tilde{\mathbf{g}}_k^{(m)\mathrm{WPE}} &= \underset{\tilde{\mathbf{g}}_k^{(m)},\boldsymbol{\lambda}_k>0}{\operatorname{argmin}} \mathcal{L}\left(\mathbf{d}_k^{(m)}, \boldsymbol{\lambda}_k\right) = \underset{\tilde{\mathbf{g}}_k^{(m)},\boldsymbol{\lambda}_k>0}{\operatorname{argmin}} -\log \mathbb{L}\left(\mathbf{d}_k^{(m)}, \boldsymbol{\lambda}_k\right) \\ &= \underset{\tilde{\mathbf{g}}_k^{(m)},\boldsymbol{\lambda}_k>0}{\operatorname{argmin}} -\sum_{t=1}^{T} \log \rho\left(d_{k,t}^{(m)}, \lambda_{k,t}\right) = \underset{\tilde{\mathbf{g}}_k^{(m)},\boldsymbol{\lambda}_k>0}{\operatorname{argmin}} \sum_{t=1}^{T} \left(\frac{\left|d_{k,t}^{(m)}\right|^2}{\lambda_{k,t}} + \log \pi \lambda_{k,t}\right) \\ &= \underset{\tilde{\mathbf{g}}_k^{(m)},\boldsymbol{\lambda}_k>0}{\operatorname{argmin}} \mathbf{d}_k^{(m)} \boldsymbol{\Lambda}_k^{-1} \mathbf{d}_k^{(m)H} + \sum_{t=1}^{T} \log \lambda_{k,t} + T \log \pi\end{aligned} \tag{2.28}$$

whereby $\boldsymbol{\Lambda}_k$ is a diagonal matrix containing the variances $\lambda_{k,t}$ of each time frame t on its diagonal. Since it is not possible to jointly minimize the cost function in eq. (2.28) with respect to the prediction filter $\tilde{\mathbf{g}}_k^{(m)}$ and the variances $\boldsymbol{\lambda}_k$ analytically an alternating optimization procedure was proposed in [15]. For this the optimization problem is divided into two subproblems, whereby either the variances $\boldsymbol{\lambda}_k$ or the prediction filter $\tilde{\mathbf{g}}_k^{(m)}$ are assumed to be fixed. Now the optimization of these two subproblems is performed alternatingly in an iterative scheme until the convergence of the dereverberated signal, which is the output. The convergence is measured by the relative convergence criterion

$$\eta = \frac{\left\|\breve{\mathbf{d}}_{k,\mathrm{cur}}^{(m)} - \breve{\mathbf{d}}_{k,\mathrm{old}}^{(m)}\right\|_2}{\left\|\breve{\mathbf{d}}_{k,\mathrm{old}}^{(m)}\right\|_2} < \eta_c \tag{2.29}$$

whereby $\breve{\mathbf{d}}_{k,\mathrm{cur}}^{(m)}$ and $\breve{\mathbf{d}}_{k,\mathrm{old}}^{(m)}$ are the dereverberated signals of the current iteration and the last iteration respectively.

2.4.2.1.1 Estimation of the Prediction Filter $\tilde{\mathbf{g}}_k^{(m)}$ In order to minimize the cost function in eq. (2.28) in respect to only the prediction filter $\tilde{\mathbf{g}}_k^{(m)}$ the variances $\boldsymbol{\lambda}_k$ are assumed to be fixed so that the cost function reduces to

$$\tilde{\mathbf{g}}_k^{(m)(i)} = \underset{\tilde{\mathbf{g}}_k^{(m)}}{\operatorname{argmin}}\, \breve{\mathbf{d}}_k^{(m)(i)} \left(\boldsymbol{\Lambda}_k^{(i)}\right)^{-1} \breve{\mathbf{d}}_k^{(m)(i)H} \quad \text{s.t.} \quad \breve{\mathbf{d}}_k^{(m)(i)} = \mathbf{y}_k^{(m)} - \tilde{\mathbf{g}}_k^{(m)H} \tilde{\mathbf{Y}}_k \tag{2.30}$$

where i is the index of the iteration. This has the following least-squares (LS) solution:

$$\boxed{\tilde{\mathbf{g}}_k^{(m)(i)} = \left(\tilde{\mathbf{R}}_{\tilde{y}\lambda,k}^{(i)}\right)^{-1} \mathbf{p}_{y\lambda,k}^{(m)\,(i)}} \tag{2.31}$$

whereby $\tilde{\mathbf{R}}_{\tilde{y}\lambda,k} = \mathbb{E}\left[\frac{\tilde{\mathbf{y}}_{k,t}\tilde{\mathbf{y}}_{k,t}^H}{\lambda_{k,t}}\right] \approx \frac{1}{T}\tilde{\mathbf{Y}}_k \boldsymbol{\Lambda}_k^{-1} \tilde{\mathbf{Y}}_k^H = \frac{1}{T}\sum_{t=1}^T \frac{\tilde{\mathbf{y}}_{k,t}\tilde{\mathbf{y}}_{k,t}^H}{\lambda_{k,t}} \in \mathbb{C}^{M(L_k-\tau_k)\times M(L_k-\tau_k)}$ is the variance-weighted noisy covariance matrix of the past frames corresponding to the late reverberation and $\mathbf{p}_{y\lambda,k}^{(m)} = \mathbb{E}\left[\frac{\tilde{\mathbf{y}}_{k,t} y_{k,t}^{(m)*}}{\lambda_{k,t}}\right] \approx \frac{1}{T}\tilde{\mathbf{Y}}_k \boldsymbol{\Lambda}_k^{-1} \mathbf{y}_k^{(m)H} = \frac{1}{T}\sum_{t=1}^T \frac{\tilde{\mathbf{y}}_{k,t} y_{k,t}^{(m)*}}{\lambda_{k,t}} \in \mathbb{C}^{M(L_k-\tau_k)}$ is the variance-weighted noisy cross-covariance vector of the past frames with the momentary frame $y_{k,t}^{(m)}$ of the reference channel m.

2.4.2.1.2 Estimation of Variances $\boldsymbol{\lambda}_k$ In the second step the cost function in eq. (2.28) is minimized with respect to the variances $\boldsymbol{\lambda}_k$ where the optimal prediction filter $\tilde{\mathbf{g}}_k^{(m)}$ is assumed to be fixed. The cost function of the subproblem is then given for each time frame t individually by:

$$\lambda_{k,t}^{(i)} = \underset{\lambda_{k,t}>0}{\operatorname{argmin}} \frac{\left|\breve{d}_{k,t}^{(m)(i-1)}\right|^2}{\lambda_{k,t}} + \log \lambda_{k,t} \tag{2.32}$$

The solution for this subproblem is

$$\lambda_{k,t}^{(i)} = \left|\breve{d}_{k,t}^{(m)(i-1)}\right|^2 \quad \Leftrightarrow \quad \boldsymbol{\lambda}_k^{(i)} = \left|\breve{\mathbf{d}}_k^{(m)(i-1)}\right|^2 \tag{2.33}$$

whereby the absolute value operator is applied elementwise. For a practical algorithm a small positive constant ε is added to prevent division by zero

$$\lambda_{k,t}^{(i)} = \left|\breve{d}_{k,t}^{(m)(i-1)} + \varepsilon\right|^2 \quad \Leftrightarrow \quad \boldsymbol{\lambda}_k^{(i)} = \left|\breve{\mathbf{d}}_k^{(m)(i-1)} + \varepsilon\right|^2 \tag{2.34}$$

2.4.2.2 MCLP Dereverberation using a General Sparse Prior

The conventional WPE utilizes a TVG model as sparse prior. However the WPE algorithm can be generalized with any circular sparse prior for the desired signal

$d_{k,t}^{(m)}$ [17], which is represented by the following general PDF

$$\rho\left(d_{k,t}^{(m)}\right) = e^{-f\left(\left|d_{k,t}^{(m)}\right|\right)} \tag{2.35}$$

It can be shown that the prior ρ is sparse when $f'(\bullet)/\bullet$ is decreasing on $\bullet \in (0, \infty)$, whereby $f'(\bullet)$ denotes the derivative of $f(\bullet)$. If this condition is fulfilled the sparse prior PDF $\rho\left(d_{k,t}^{(m)}\right)$ can be represented as a maximization over scaled Gaussians as

$$\rho\left(d_{k,t}^{(m)}\right) = \max_{\lambda_{k,t}>0} \mathcal{N}_{\mathbb{C}}\left(d_{k,t}^{(m)}; 0; \lambda_{k,t}\right) \zeta\left(\lambda_{k,t}\right) \tag{2.36}$$

Hereby the scaling function $\zeta(\bullet)$ can be interpreted as hyperprior on the variance $\lambda_{k,t}$. Similarly to eq. (2.28) a LH function for the general sparse prior can be formulated as

$$\mathbb{L}\left(\mathbf{d}_k^{(m)}, \boldsymbol{\lambda}_k\right) = \prod_{t=1}^{T} \rho\left(d_{k,t}^{(m)}, \lambda_{k,t}\right) = \prod_{t=1}^{T} \max_{\lambda_{k,t}>0} \mathcal{N}_{\mathbb{C}}\left(d_{k,t}^{(m)}; 0; \lambda_{k,t}\right) \zeta\left(\lambda_{k,t}\right) \tag{2.37}$$

and likewise also the nLLH function of the general sparse prior, which can be seen as the cost function, can be reformulated in a way that it is to be minimized over the prediction filter $\tilde{\mathbf{g}}_k^{(m)}$ and the variances $\boldsymbol{\lambda}_k$ in order to find the optimal filter as

$$\begin{aligned}
\tilde{\mathbf{g}}_k^{(m)\text{WPE}} &= \underset{\tilde{\mathbf{g}}_k^{(m)}}{\operatorname{argmin}} \mathcal{L}\left(\mathbf{d}_k^{(m)}, \boldsymbol{\lambda}_k\right) = \underset{\tilde{\mathbf{g}}_k^{(m)}}{\operatorname{argmin}} \sum_{t=1}^{T} -\log\rho\left(d_{k,t}^{(m)}, \lambda_{k,t}\right) \\
&= \underset{\tilde{\mathbf{g}}_k^{(m)}}{\operatorname{argmin}} \sum_{t=1}^{T} - \max_{\lambda_{k,t}>0} \log\mathcal{N}_{\mathbb{C}}\left(d_{k,t}^{(m)}; 0; \lambda_{k,t}\right) \zeta\left(\lambda_{k,t}\right) \\
&= \underset{\tilde{\mathbf{g}}_k^{(m)}}{\operatorname{argmin}} \sum_{t=1}^{T} \min_{\lambda_{k,t}>0} - \log\mathcal{N}_{\mathbb{C}}\left(d_{k,t}^{(m)}; 0; \lambda_{k,t}\right) \zeta\left(\lambda_{k,t}\right) \\
&= \underset{\tilde{\mathbf{g}}_k^{(m)}, \boldsymbol{\lambda}_k}{\operatorname{argmin}} \sum_{t=1}^{T} - \log\mathcal{N}_{\mathbb{C}}\left(d_{k,t}^{(m)}; 0; \lambda_{k,t}\right) \zeta\left(\lambda_{k,t}\right) \\
&= \underset{\tilde{\mathbf{g}}_k^{(m)}, \boldsymbol{\lambda}_k}{\operatorname{argmin}} \sum_{t=1}^{T} \left(\frac{\left|d_{k,t}^{(m)}\right|^2}{\lambda_{k,t}} + \log\pi\lambda_{k,t} - \log\zeta\left(\lambda_{k,t}\right) \right)
\end{aligned} \tag{2.38}$$

The joint optimization for this cost function is again not possible analytically as described in section 2.4.2.1 for the conventional WPE. However a similar alternating iterative optimization scheme can be utilized here, whereby the prediction filter $\tilde{\mathbf{g}}_k^{(m)}$ and the variances $\boldsymbol{\lambda}_k$ are updated in separate steps.

2.4.2.2.1 Estimation of the Prediction Filter $\tilde{\mathbf{g}}_k^{(m)}$ The update of the prediction filter $\tilde{\mathbf{g}}_k^{(m)}$ is hereby completely equal to the conventional method in paragraph 2.4.2.1.1 given by eq. (2.31), since the cost function in eq. (2.38) reduces to the same expression as eq. (2.30), if the variances $\boldsymbol{\lambda}_k$ are assumed to be fixed.

2.4.2.2.2 Estimation of Variances $\boldsymbol{\lambda}_k$ Assuming the prediction filter $\tilde{\mathbf{g}}_k^{(m)}$ to be fixed enables reformulation of the cost function for the variance update for each time frame t individually as

$$\lambda_{k,t}^{(i)} = \underset{\lambda_{k,t}>0}{\operatorname{argmin}} \frac{\left|\breve{d}_{k,t}^{(m)\,(i-1)}\right|^2}{\lambda_{k,t}} + \log \pi\lambda_{k,t} - \log \zeta\left(\lambda_{k,t}\right) \tag{2.39}$$

whereby the scaling function $\zeta(\lambda_{k,t})$ of the variance introduces an additional term comparing to the conventional cost function in paragraph 2.4.2.1.2. The solution for a general sparse prior to this subproblem is given by

$$\boxed{\lambda_{k,t}^{(i)} = \frac{2\left|\breve{d}_{k,t}^{(m)\,(i-1)}\right|}{f'\left(\left|\breve{d}_{k,t}^{(m)\,(i-1)}\right|\right)}} \tag{2.40}$$

2.4.2.3 MCLP Dereverberation using a CGG Sparse Prior

One example of a general sparse prior is the CGG sparse prior [17], which is more general than the TVG model. The prior is given by

$$\rho\left(d_{k,t}^{(m)}\right) = \frac{p}{2\pi\beta\Gamma\left(2/p\right)} e^{-\frac{\left|d_{k,t}^{(m)}\right|^p}{\beta^{p/2}}} \tag{2.41}$$

where $\beta \in \mathbb{R}_{>0}$ is its scale parameter, $p \in \left]0, 2\right]$ is its shape parameter and $\Gamma\left(\bullet\right)$ is the gamma function. As seen in fig. 2.3 for $p = 2$ the CGG prior equals a Gaussian and for smaller values of p it is seen to be super Gaussian i.e. sparse. When written in the format of eq. (2.35) the function $f(\bullet)$ is given by

$$f\left(\left|d_{k,t}^{(m)}\right|\right) = \frac{\left|d_{k,t}^{(m)}\right|^p}{\beta^{p/2}} - \log \frac{p}{2\pi\beta\Gamma\left(2/p\right)} \tag{2.42}$$

so that its derivative can be stated as

$$f'\left(\left|d_{k,t}^{(m)}\right|\right) = \frac{p\left|d_{k,t}^{(m)}\right|^{p-1}}{\beta^{p/2}} \tag{2.43}$$

By inserting this derivative into the variance update of the general sparse prior in eq. (2.40) the following variance update is obtained:

$$\lambda_{k,t}^{(i)} = \frac{2\beta^{p/2}}{p} \left| \breve{d}_{k,t}^{(m)(i-1)} \right|^{2-p} \tag{2.44}$$

However since the estimation of the prediction filter $\tilde{\mathbf{g}}_k^{(m)}$ given by eq. (2.31) is invariant to a scaling of the variances $\boldsymbol{\lambda}_k$, the variance update can be reduced to

$$\boxed{\lambda_{k,t}^{(i)} = \left| \breve{d}_{k,t}^{(m)(i-1)} + \varepsilon \right|^{2-p}} \tag{2.45}$$

where for a practical algorithm a small positive constant ε is added to avoid division by zero.

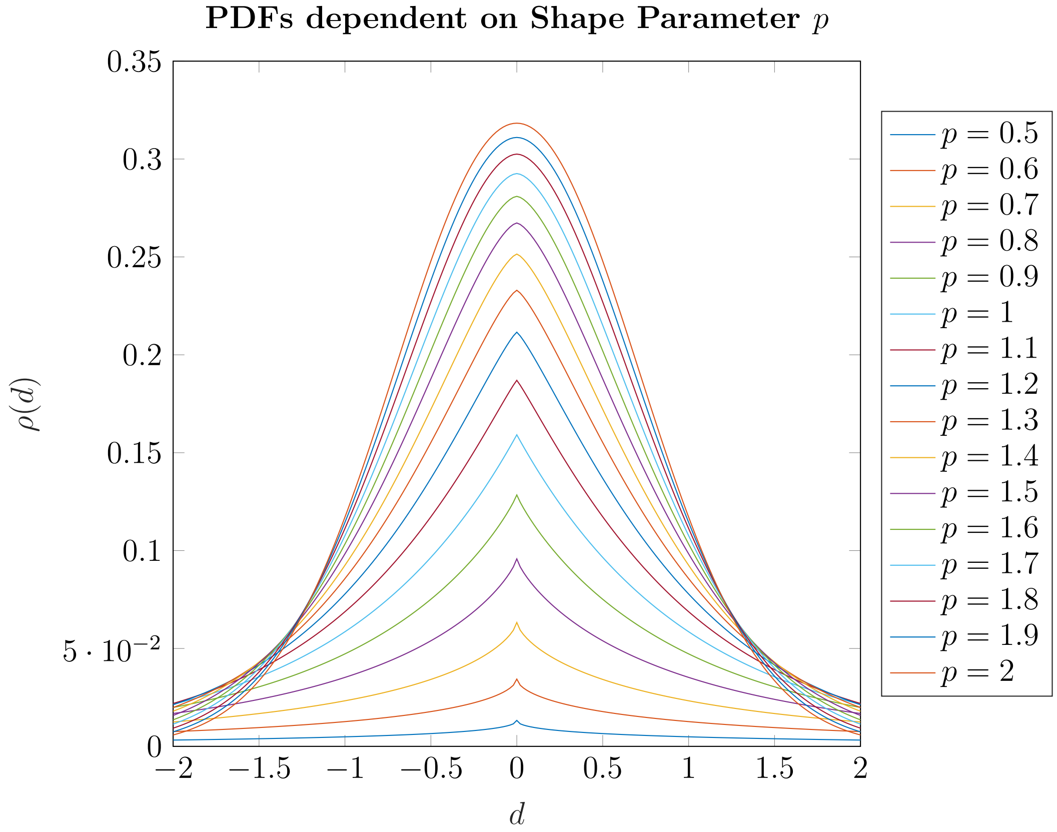

Figure 2.3: CGG-PDFs according to eq. (2.41) dependent on the shape parameter p. The scale parameter is chosen to be $\beta = 1$. The plot only shows the PDF value for real STFT coefficients $d_{k,t}^{(m)}$. However the actual PDF is circular over the complex plane.

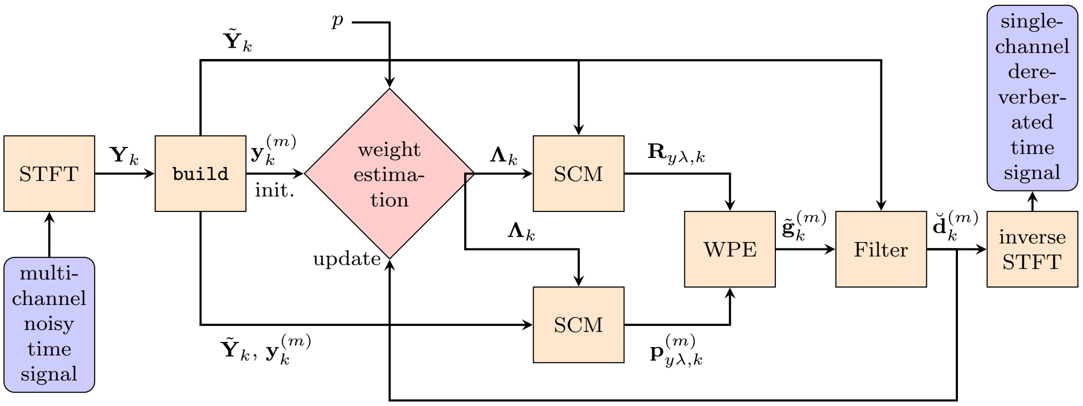
STFT
$\mathbf{Y}_k$
build
$\mathbf{y}_k^{(m)}$
init.
$\tilde{\mathbf{Y}}_k$
p
weight estimation
update
$\mathbf{\Lambda}_k$
$\mathbf{\Lambda}_k$
SCM
SCM
$\mathbf{R}_{y\lambda,k}$
$\mathbf{p}_{y\lambda,k}^{(m)}$
$\tilde{\mathbf{Y}}_k$, $\mathbf{y}_k^{(m)}$
WPE
$\tilde{\mathbf{g}}_k^{(m)}$
Filter
$\check{\mathbf{d}}_k^{(m)}$
inverse STFT
single-channel dereverberated time signal
multi-channel noisy time signal

Figure 2.4: Flow chart of MISO-WPE.

2.4.2.4 Reformulation using the ℓ_p-Norm

It is possible to reformulate the derivation of the WPE algorithm with a cost function, which we show is equivalent to the LH function using the CGG sparse prior, as [17]

$$\mathcal{C}\left(\mathbf{d}_k^{(m)}\right) = \left\|\mathbf{d}_k^{(m)}\right\|_p^p \tag{2.46}$$

where $\|\bullet\|_p = \sqrt[p]{\sum_{n=1}^{N}|\bullet_n|^p}$ denotes the ℓ_p-norm[1] of a vector. In order to solve the following resulting optimization problem

$$\tilde{\mathbf{g}}_k^{(m)} = \underset{\tilde{\mathbf{g}}_k^{(m)}}{\text{argmin}}\left\|\mathbf{d}_k^{(m)}\right\|_p^p \tag{2.47}$$

and to show the equivalency to the CGG sparse prior formulation in section 2.4.2.3 the ℓ_p-norm minimization is replaced with a series of ℓ_2-norm minimization subproblems, which have an LS solution. The new cost function of these ℓ_2-norm minimization subproblems is then given by

$$\tilde{\mathbf{g}}_k^{(m)(i)} = \underset{\tilde{\mathbf{g}}_k^{(m)}}{\text{argmin}}\, \mathbf{d}_k^{(m)(i)}\left(\mathbf{\Lambda}_k^{(i)}\right)^{-1}\mathbf{d}_k^{(m)(i)H} \tag{2.48}$$

which is already known from eq. (2.30). Although in this context $\mathbf{\Lambda}_k$ denotes the inverse of the optimization weights, which enables the ℓ_2-norm minimization subproblems, the same nomenclature as for the variances in the former derivations is used, because they are equivalent. Therefore also the update of the weights is given by the variance update in eq. (2.45). A flow chart and the practical MISO-WPE algorithm are presented in fig. 2.4 and algorithm 3 respectively.

[1] For $p < 1$ this is actually not a norm, since the triangle inequality is violated. However here it is still referred to as ℓ_p-norm.

```
Algorithm 3: MISO-WPE dereverberation (batch)

input      : batch-matrix of multi-channel noisy microphone signal $\mathbf{Y}_k$ $\forall$ $k$
parameters: reference channel $m$, frequency dependent prediction delay $\tau_k$,
             frequency dependent prediction filter length $L_k$, shape parameter $p$,
             regularization parameter $\varepsilon$, maximal number of iterations of the
             alternating optimization $I_{max}$, convergence tolerance $\eta_c$
functions  : constructing convolutional signal matrix build (•), constructing
             diagonal matrix from vector diagMat (•)
output     : batch-vector of single-channel estimated dereverberated signal $\breve{\mathbf{d}}_k^{(m)}$ $\forall$ $k$

foreach $k \in \{1, 2, \ldots, K\}$ do          // process each frequency subband $k$ individually
    $\mathbf{y}_k^{(m)} = \mathbf{e}_M^{(m)T} \mathbf{Y}_k$          // extracts batch-vector of single-channel convolutional signal
    $\tilde{\mathbf{Y}}_k = \mathtt{build}\left(\mathbf{Y}_k; \tau_k, L_k\right)$          // builds convolutional signal matrix with past frames
    $\mathbf{\Lambda}_k = \mathtt{diagMat}\left(\left|\mathbf{y}_k^{(m)}\right|^2 + \varepsilon\right)^{1-\frac{p}{2}}$          // initialize weights using the noisy signal
    $\breve{\mathbf{d}}_{k,\mathrm{old}}^{(m)} = \mathbf{y}_k^{(m)}$          // initialize $\breve{\mathbf{d}}_{k,\mathrm{old}}^{(m)}$ with the noisy signal
    for $i \leftarrow 1$ to $I_{max}$ do          // iterations of alternating optimization
        $\tilde{\mathbf{R}}_{\tilde{y}\lambda,k} = \frac{1}{T} \tilde{\mathbf{Y}}_k \mathbf{\Lambda}_k^{-1} \tilde{\mathbf{Y}}_k^H$          // estimate weighted noisy covariance matrix by SCM
        $\mathbf{p}_{y\lambda,k}^{(m)} = \frac{1}{T} \tilde{\mathbf{Y}}_k \mathbf{\Lambda}_k^{-1} \mathbf{y}_k^{(m)H}$          // estimate weighted noisy cross-covariance vector by SCM
        $\tilde{\mathbf{g}}_k^{(m)} = \tilde{\mathbf{R}}_{\tilde{y}\lambda,k}^{-1} \mathbf{p}_{y\lambda,k}^{(m)}$          // estimate reverberation filter vector
        $\breve{\mathbf{d}}_k^{(m)} = \mathbf{y}_k^{(m)} - \tilde{\mathbf{g}}_k^{(m)H} \tilde{\mathbf{Y}}_k$          // dereverberation by subtracting estimated reverberation
        if $\frac{\left\|\breve{\mathbf{d}}_k^{(m)} - \breve{\mathbf{d}}_{k,\mathrm{old}}^{(m)}\right\|_2}{\left\|\breve{\mathbf{d}}_{k,\mathrm{old}}^{(m)}\right\|_2} < \eta_c$ then          // relative convergence criterion
            break          // breaks the for loop of the alternating optimization
        else
            $\mathbf{\Lambda}_k = \mathtt{diagMat}\left(\left|\breve{\mathbf{d}}_k^{(m)}\right|^2 + \varepsilon\right)^{1-\frac{p}{2}}$          // compute scaled signal power weights
            $\breve{\mathbf{d}}_{k,\mathrm{old}}^{(m)} = \breve{\mathbf{d}}_k^{(m)}$          // store output for convergence criterion of next iteration
```

2.4.2.5 MIMO-WPE Dereverberation

For many applications a MIMO version of the WPE algorithm is very useful, e.g. it enables cascade beamforming for additional denoising of the dereverberated signal. The following derivations are based on [18]. A MIMO filter matrix $\tilde{\mathbf{G}}_k = \left[\tilde{\mathbf{g}}_k^{(1)}, \tilde{\mathbf{g}}_k^{(2)}, \ldots, \tilde{\mathbf{g}}_k^{(M)}\right] \in \mathbb{C}^{M(L_k - \tau_k) \times M}$ is introduced, which contains the prediction filter vectors $\tilde{\mathbf{g}}_k^{(m)}$ for every reference channel m in its columns. The subtraction of the predicted multi-channel late reverberation $\mathbf{r}_{k,t}$ from the multi-channel noisy signal $\mathbf{y}_{k,t}$ can be formulated similarly to eq. (2.22) as

$$\breve{\mathbf{d}}_{k,t} = \mathbf{y}_{k,t} - \mathbf{r}_{k,t} = \mathbf{y}_{k,t} - \underbrace{\tilde{\mathbf{G}}_k{}^H \tilde{\mathbf{y}}_{k,t}}_{\mathbf{r}_{k,t}} = \bar{\mathbf{G}}_k{}^H \bar{\mathbf{y}}_{k,t} \tag{2.49}$$

with

$$\bar{\mathbf{G}}_k = \begin{bmatrix} \mathbf{I}_M \\ -\tilde{\mathbf{G}}_k \end{bmatrix} = \left[\bar{\mathbf{g}}_k^{(1)}, \bar{\mathbf{g}}_k^{(2)}, \ldots, \bar{\mathbf{g}}_k^{(M)}\right] \in \mathbb{C}^{M(L_k-\tau_k+1)\times M} \tag{2.50}$$

Using the batch matrices of the noisy signal $\mathbf{Y}_k = [\mathbf{y}_{k,1}, \mathbf{y}_{k,2}, \ldots, \mathbf{y}_{k,T}] \in \mathbb{C}^{M\times T}$ and the dereverberated signal $\breve{\mathbf{D}}_k = \left[\breve{\mathbf{d}}_{k,1}, \breve{\mathbf{d}}_{k,2}, \ldots, \breve{\mathbf{d}}_{k,T}\right] \in \mathbb{C}^{M\times T}$ leads to the compact MIMO-WPE formulation

$$\boxed{\breve{\mathbf{D}}_k = \bar{\mathbf{G}}_k^H \mathbf{Y}_k} \tag{2.51}$$

To determine the optimal prediction filter matrix $\tilde{\mathbf{G}}_k$ an extension of the ℓ_p-norm optimization can be formulated using the mixed $\ell_{\mathbf{\Phi};2,p}$-norm as proposed in [18] given by

$$\mathcal{C}\left(\mathbf{D}_k\right) = \|\mathbf{D}_k\|^p_{\mathbf{\Phi};2,p} = \sum_{t=1}^{T} \|\mathbf{d}_{k,t}\|^p_{\mathbf{\Phi};2} = \sum_{t=1}^{T} \left(\sqrt{\mathbf{d}_{k,t}^H \mathbf{\Phi}^{-1} \mathbf{d}_{k,t}}\right)^p \tag{2.52}$$

whereby the matrix $\mathbf{\Phi}$ models the spatial correlations between the channels. The ℓ_p-norm optimization problem of this cost function can be approximated with a series of weighted ℓ_2-norm subproblems similarly to the MISO-WPE in section 2.4.2.3 as

$$\sum_{t=1}^{T} \|\mathbf{d}_{k,t}\|^p_{\mathbf{\Phi};2} \approx \sum_{t=1}^{T} \frac{\left\|\mathbf{d}_{k,t}^{(i)}\right\|^2_{\mathbf{\Phi};2}}{\lambda_{k,t}^{(i)}} = \mathtt{trace}\left(\mathbf{\Lambda}_k^{(i)-1} \mathbf{D}_k^{(i)H} \mathbf{\Phi}^{-1} \mathbf{D}_k^{(i)}\right) \tag{2.53}$$

with the weights $\lambda_{k,t}^{(i)}$ selected so that eq. (2.53) is a first-order approximation of the corresponding $\ell_{\mathbf{\Phi};2,p}$-norm cost function:

$$\boxed{\lambda_{k,t}^{(i)} = \left\|\breve{\mathbf{d}}_{k,t}^{(i-1)}\right\|^{2-p}_{\mathbf{\Phi};2}} \tag{2.54}$$

The subproblem of estimating the prediction filter matrix $\tilde{\mathbf{G}}_k$ is then given by

$$\tilde{\mathbf{G}}_k^{(i)} = \underset{\tilde{\mathbf{G}}_k}{\operatorname{argmin}}\, \mathtt{trace}\left(\left(\mathbf{Y}_k - \tilde{\mathbf{G}}_k^H \tilde{\mathbf{Y}}_k\right) \mathbf{\Lambda}_k^{(i)-1} \left(\mathbf{Y}_k - \tilde{\mathbf{G}}_k^H \tilde{\mathbf{Y}}_k\right)^H \mathbf{\Phi}^{-1}\right) \tag{2.55}$$

and it has the following LS solution:

$$\boxed{\tilde{\mathbf{G}}_k^{(i)} = \tilde{\mathbf{R}}_{\tilde{y}\lambda,k}^{(i)-1} \mathbf{P}_{y\lambda,k}^{(i)}} \tag{2.56}$$

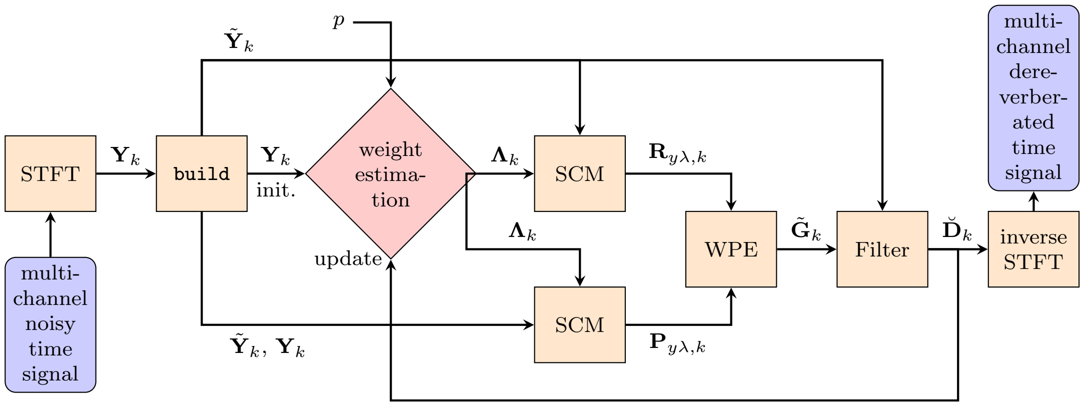

Figure 2.5: Flow chart of MIMO-WPE.

whereby $\mathbf{P}_{y\lambda,k} = \mathbb{E}\left[\frac{\tilde{\mathbf{y}}_{k,t}\mathbf{y}_{k,t}^H}{\lambda_{k,t}}\right] = \left[\mathbf{p}_{y,k}^{(1)}, \mathbf{p}_{y,k}^{(2)}, \ldots, \mathbf{p}_{y,k}^{(M)}\right] \approx \frac{1}{T}\tilde{\mathbf{Y}}_k\mathbf{\Lambda}_k^{-1}\mathbf{Y}_k^H = \frac{1}{T}\sum_{t=1}^{T}\frac{\tilde{\mathbf{y}}_{k,t}\mathbf{y}_{k,t}^H}{\lambda_{k,t}} \in \mathbb{C}^{M(L_k-\tau_k)\times M}$ is the weighted multi-channel cross-covariance matrix of the past frames with the momentary frame t. The convergence of the dereverberated signal is measured by the relatively convergence criterion

$$\eta = \frac{\left\|\breve{\mathbf{D}}_{k,\text{cur}} - \breve{\mathbf{D}}_{k,\text{old}}\right\|_{Fro}}{\left\|\breve{\mathbf{D}}_{k,\text{old}}\right\|_{Fro}} < \eta_c \tag{2.57}$$

whereby $\breve{\mathbf{D}}_{k,\text{cur}}$ and $\breve{\mathbf{D}}_{k,\text{old}}$ are the dereverberated signals of the current iteration and the last iteration respectively and the Frobenius norm of a matrix is defined as $\|\bullet\|_{\text{Fro}} = \sqrt{\sum_{n_1=1}^{N_1}\sum_{n_2=1}^{N_2}|\bullet_{n_1,n_2}|^2}$. An overview of the complete workflow of MIMO-WPE is presented in fig. 2.5 and algorithm 4.

2.4.2.5.1 Group Sparsity As proposed in [18] the matrix $\mathbf{\Phi}$ is understood to model the spatial (within-group) correlation of the multi-channel desired signal. Hereby a group consists of the channels $m = 1, 2, \ldots, M$. The update of the so called group sparsity $\mathbf{\Phi}$ can be given by

$$\mathbf{\Phi}^{(i)} = \frac{1}{T}\sum_{t=1}^{T}\frac{\breve{\mathbf{d}}_{k,t}^{(i)}\breve{\mathbf{d}}_{k,t}^{(i)H}}{\lambda_{k,t}^{(i)}} = \frac{1}{T}\breve{\mathbf{D}}_k^{(i)}\mathbf{\Lambda}_k^{(i)-1}\breve{\mathbf{D}}_k^{(i)H} \tag{2.58}$$

However preliminary results of simulated experiments indicated that this group sparsity update is not stable and did not lead to improvements of speech quality. Therefore in all of the following derivations the group sparsity is neglected by assuming that there are no spatial correlations between the microphone channels. So the group sparsity correlation is set to be an identity matrix $\mathbf{\Phi} = \mathbf{I}_M$.

Algorithm 4: MIMO-WPE dereverberation (batch)

input : batch-matrix of multi-channel noisy microphone signal $\mathbf{Y}_k \ \forall \ k$

parameters: reference channel m, frequency dependent prediction delay τ_k, frequency dependent prediction filter length L_k, shape parameter p, regularization parameter ε, maximal number of iterations of the alternating optimization I_{max}, convergence tolerance η_c

functions : constructing convolutional signal matrix `build` $(\bullet)$, constructing diagonal matrix from vector `diagMat` $(\bullet)$

output : batch-matrix of multi-channel estimated dereverberated signal $\breve{\mathbf{D}}_k \ \forall \ k$

```
foreach $k \in \{1, 2, \ldots, K\}$ do    // process each frequency subband $k$ individually
    $\tilde{\mathbf{Y}}_k \leftarrow \texttt{build}\,(\mathbf{Y}_k; \tau_k, L_k)$    // builds convolutional signal matrix with past frames
    foreach $t \in \{1, 2, \ldots, T\}$ do    // process each time frame $t$ individually
        $\lambda_{k,t} = \left(\|\mathbf{y}_{k,t}\|^2_{\mathbf{\Phi};2} + \varepsilon\right)^{1-\frac{p}{2}}$    // initialize weights by mixed norm of the noisy signal
    $\mathbf{\Lambda}_k = \texttt{diagMat}\,(\boldsymbol{\lambda}_k = [\lambda_{k,1}, \lambda_{k,2}, \ldots, \lambda_{k,T}])$
    $\breve{\mathbf{D}}_{k,\text{old}} = \mathbf{Y}_k$    // initialize $\breve{\mathbf{D}}_{k,\text{old}}$ with the noisy signal
    for $i \leftarrow 1$ to $I_{max}$ do    // iterations of alternating optimization
        $\tilde{\mathbf{R}}_{\tilde{y}\lambda,k} = \frac{1}{T}\tilde{\mathbf{Y}}_k\mathbf{\Lambda}_k^{-1}\tilde{\mathbf{Y}}_k^H$    // estimate weighted noisy covariance matrix by SCM
        $\mathbf{P}_{y\lambda,k} = \frac{1}{T}\tilde{\mathbf{Y}}_k\mathbf{\Lambda}_k^{-1}\mathbf{Y}_k^H$    // estimate weighted noisy cross-covariance matrix by SCM
        $\tilde{\mathbf{G}}_k = \tilde{\mathbf{R}}_{\tilde{y}\lambda,k}^{-1}\mathbf{P}_{y\lambda,k}$    // estimate reverberation filter matrix
        $\breve{\mathbf{D}}_k = \mathbf{Y}_k - \tilde{\mathbf{G}}_k^H\tilde{\mathbf{Y}}_k$    // dereverberation by subtracting estimated reverberation
        if $\frac{\|\breve{\mathbf{D}}_k - \breve{\mathbf{D}}_{k,\text{old}}\|_{\text{Fro}}}{\|\breve{\mathbf{D}}_{k,\text{old}}\|_{\text{Fro}}} < \eta_c$ then    // relative convergence criterion
            break    // breaks the for loop of the alternating optimization
        else
            foreach $t \in \{1, 2, \ldots, T\}$ do    // process each time frame $t$ individually
                $\lambda_{k,t} = \left(\left\|\breve{\mathbf{d}}_{k,t}\right\|^2_{\mathbf{\Phi};2} + \varepsilon\right)^{1-\frac{p}{2}}$    // updating weights by mixed norm of $\breve{\mathbf{d}}_{k,t}$
            $\mathbf{\Lambda}_k = \texttt{diagMat}\,(\boldsymbol{\lambda}_k = [\lambda_{k,1}, \lambda_{k,2}, \ldots, \lambda_{k,T}])$
            $\breve{\mathbf{D}}_{k,\text{old}} = \breve{\mathbf{D}}_k$    // store output for convergence criterion of next iteration
```

2.5 WPD Convolutional Beamforming

The main idea of WPD unified dereverberation and denoising is to set up an algorithm with a MIMO-WPE dereverberation stage (described in section 2.4.2.5) and an additional MPDR beamformer stage (described in section 2.2). While cascade systems of the algorithms are already widely known, the novel idea proposed by [1, 29] is joint optimization of the two algorithms in order to improve both dereverberation and denoising performance simultaneously. For this the full signal model in eq. (2.1) is utilized and a filter vector of the form $\bar{\mathbf{h}}_k^{(m)} = \bar{\mathbf{G}}_k\mathbf{q}_k^{(m)} \in \mathbb{C}^{M(L_k-\tau_k+1)}$ performs MISO joint dereverberation and denoising on the noisy signal $\bar{\mathbf{y}}_{k,t}$ as

$$z_{k,t}^{(m)} = \bar{\mathbf{h}}_k^{(m)H}\bar{\mathbf{y}}_{k,t} \tag{2.59}$$

where $z_{k,t}^{(m)}$ is the single-channel filtered signal of this convolutional beamformer of the time frame t. This can also be written in a batch formulation as

$$\mathbf{z}_k^{(m)} = \bar{\mathbf{h}}_k^{(m)H} \bar{\mathbf{Y}}_k \tag{2.60}$$

whereby $\mathbf{z}_k^{(m)} = \left[z_{k,1}^{(m)}, z_{k,2}^{(m)}, \ldots, z_{k,T}^{(m)} \right] \in \mathbb{C}^{1 \times T}$ is the batch-vector of the filtered signal.

2.5.1 Filter Optimization

In order to optimize the filter $\bar{\mathbf{h}}_k^{(m)}$ a LH function can be set up similarly to eq. (2.28)

$$\begin{aligned}
\bar{\mathbf{h}}_k^{(m)\text{WPD}} &= \underset{\bar{\mathbf{h}}_k^{(m)}, \boldsymbol{\lambda}_k > 0}{\operatorname{argmin}} \sum_{t=1}^{T} \left(\frac{\left| z_{k,t}^{(m)} \right|^2}{\lambda_{k,t}} + \log \pi \lambda_{k,t} \right) \\
&= \underset{\bar{\mathbf{h}}_k^{(m)}, \boldsymbol{\lambda}_k > 0}{\operatorname{argmin}} \ \mathbf{z}_k^{(m)} \boldsymbol{\Lambda}_k^{-1} \mathbf{z}_k^{(m)H} + \sum_{t=1}^{T} \log \lambda_{k,t} + T \log \pi \\
&= \underset{\bar{\mathbf{h}}_k^{(m)}, \boldsymbol{\lambda}_k > 0}{\operatorname{argmin}} \ \bar{\mathbf{h}}_k^{(m)H} \bar{\mathbf{Y}}_k \boldsymbol{\Lambda}_k^{-1} \bar{\mathbf{Y}}_k^H \bar{\mathbf{h}}_k^{(m)} + \sum_{t=1}^{T} \log \lambda_{k,t} + T \log \pi \\
&= \underset{\bar{\mathbf{h}}_k^{(m)}, \boldsymbol{\lambda}_k > 0}{\operatorname{argmin}} \ \bar{\mathbf{h}}_k^{(m)H} \mathbf{R}_{y\lambda,k} \bar{\mathbf{h}}_k^{(m)} + \sum_{t=1}^{T} \log \lambda_{k,t} + T \log \pi
\end{aligned} \tag{2.61}$$

The optimization over this cost function does not have any analytic solution so that similarly to section 2.4.2.1 an iterative optimization scheme is utilized. Hereby first the variances are optimized by keeping the WPD filter $\bar{\mathbf{h}}_k^{(m)}$ fixed and in a second step the variances are fixed to optimize only over the beamforming filter $\bar{\mathbf{h}}_k^{(m)}$. These two optimization stages are repeated iteratively until the resulting beamformed signal has converged.

2.5.1.1 Estimation of Beamforming Filter $\bar{\mathbf{h}}_k^{(m)}$

The last two terms of eq. (2.61) vanish if the variances are fixed. However the distortionless constraint known from an MPDR needs to be considered so that the cost function of this subproblem becomes

$$\bar{\mathbf{h}}_k^{(m)(i)} = \underset{\bar{\mathbf{h}}_k^{(m)}}{\operatorname{argmin}} \bar{\mathbf{h}}_k^{(m)H} \mathbf{R}_{y\lambda,k}^{(i)} \bar{\mathbf{h}}_k^{(m)} \quad \text{s.t.} \quad \bar{\mathbf{h}}_k^{(m)H} \bar{\mathbf{v}}_k^{(m)} = 1 \tag{2.62}$$

Here $\bar{\mathbf{v}}_k^{(m)}$ is the RTF vector extended with $M\,(L_k - \tau_k)$ zeros corresponding to the entries of the filter $\bar{\mathbf{h}}_k^{(m)}$, which correspond to the late reverberation, e.g. perform the WPE dereverberation part. The extended RTF vector then looks like this:

$$\bar{\mathbf{v}}_k^{(m)} = \left[\tilde{\mathbf{v}}_k^{(m)},\ \underbrace{0, 0, \dots, 0}_{M(L_k-\tau_k)\text{ zeros}} \right]^T \in \mathbb{C}^{M(L_k-\tau_k+1)} \tag{2.63}$$

The solution to this subproblem can be obtained using the method of Lagrange multipliers and is equivalent to the widely known MPDR beamformer. However the weighting matrix $\mathbf{\Lambda}_k$ containing the variances and the past frames within $\bar{\mathbf{Y}}_k$ are novel and enable dereverberation in parallel to the denoising usually performed by an MPDR. The solution of this subproblem is then given by

$$\boxed{\bar{\mathbf{h}}_k^{(m)(i)} = \frac{\bar{\mathbf{R}}_{\bar{y}\lambda,k}^{(i)-1}\bar{\mathbf{v}}_k^{(m)}}{\bar{\mathbf{v}}_k^{(m)H}\bar{\mathbf{R}}_{\bar{y}\lambda,k}^{(i)-1}\bar{\mathbf{v}}_k^{(m)}}} \tag{2.64}$$

2.5.1.2 Estimation of Variances $\lambda_{k,t}$

If the beamforming filter $\bar{\mathbf{h}}_k^{(m)}$ is fixed eq. (2.61) can be used to update the variances $\lambda_{k,t}$ as

$$\boldsymbol{\lambda}_k^{(i)} = \underset{\boldsymbol{\lambda}_k > 0}{\operatorname{argmin}}\, \bar{\mathbf{h}}_k^{(m)(i-1)H}\mathbf{R}_{y\lambda,k}\bar{\mathbf{h}}_k^{(m)(i-1)} + \sum_{t=1}^{T}\log\lambda_{k,t} + T\log\pi \tag{2.65}$$

The solution to this subproblem is equivalent to the variance update of MISO-WPE in eq. (2.33) given by

$$\boxed{\lambda_{k,t}^{(i)} = \left| z_{k,t}^{(m)(i-1)} \right|^2 \quad \Leftrightarrow \quad \boldsymbol{\lambda}_k^{(i)} = \left| \mathbf{z}_k^{(m)(i-1)} \right|^2} \tag{2.66}$$

whereby the absolute value operator is applied elementwise. For a practical algorithm a small positive constant ε is added to prevent division by zero

$$\lambda_{k,t}^{(i)} = \left| z_{k,t}^{(m)(i-1)} + \varepsilon \right|^2 \quad \Leftrightarrow \quad \boldsymbol{\lambda}_k^{(i)} = \left| \mathbf{z}_k^{(m)(i-1)} + \varepsilon \right|^2 \tag{2.67}$$

2.5.2 Factorized MISO-WPD

To further understand the structure of the WPD algorithm, which performs unified dereverberation and noise reduction, the convolutional filter $\bar{\mathbf{h}}_k^{(m)}$ in eq. (2.60) can be factorized into a dereverberation filter matrix $\bar{\mathbf{G}}_k$ and a weighted minimum power distortionless response (wMPDR) beamformer filter vector $\mathbf{q}_k^{(m)}$ as [30]

$$\bar{\mathbf{h}}_k^{(m)} = \bar{\mathbf{G}}_k \mathbf{q}_k^{(m)} \quad \Rightarrow \quad \mathbf{z}_k^{(m)} = \bar{\mathbf{h}}_k^{(m)H} \bar{\mathbf{Y}}_k = \mathbf{q}_k^{(m)H} \underbrace{\bar{\mathbf{G}}_k^H \bar{\mathbf{Y}}_k}_{\breve{\mathbf{D}}_k} = \mathbf{q}_k^{(m)H} \breve{\mathbf{D}}_k \tag{2.68}$$

The optimization procedure, which is similar as in section 2.5.1.1, of this factorized version of the beamforming filter can also be factorized. However the update of the variances according to section 2.5.1.2 remains the same.

2.5.2.1 Filter Optimization

The cost function $\mathcal{C}\left(\bar{\mathbf{h}}_k^{(m)} = \bar{\mathbf{G}}_k \mathbf{q}_k^{(m)}\right)$ of the beamforming filter subproblems from eq. (2.62) can then be formulated as

$$\bar{\mathbf{h}}_k^{(m)(i)} = \underset{\bar{\mathbf{h}}_k^{(m)}}{\operatorname{argmin}}\, \mathbf{q}_k^{(m)H} \bar{\mathbf{G}}_k^H \bar{\mathbf{R}}_{\bar{y}\lambda,k}^{(i)} \bar{\mathbf{G}}_k \mathbf{q}_k^{(m)} \quad \text{s.t.} \quad \mathbf{q}_k^{(m)H} \bar{\mathbf{G}}_k^H \bar{\mathbf{v}}_k^{(m)} = 1 \tag{2.69}$$

This cost function can be separated into two steps, whereby in the first step it is optimized with respect to $\bar{\mathbf{G}}_k$ without the distortionless constraint from the beamformer, but with a structural constraint for $\bar{\mathbf{G}}_k$ in order to keep the direct signal and early reflections. In the second step the resulting filtered signal $\breve{\mathbf{D}}_k$ is used to optimize the remaining cost function with the distortionless constraint with respect to $\mathbf{q}_k^{(m)}$.

2.5.2.1.1 Estimation of Dereverberation Filter Matrix $\bar{\mathbf{G}}_k$ The cost function for the dereverberation step, which is performed at first, with the structural constraint of the dereverberation matrix $\bar{\mathbf{G}}_k$, is given by

$$\bar{\mathbf{G}}_k^{(i)} = \underset{\bar{\mathbf{G}}_k}{\operatorname{argmin}}\, \mathbf{q}_k^{(m)(i-1)H} \bar{\mathbf{G}}_k^H \bar{\mathbf{R}}_{\bar{y}\lambda,k}^{(i)} \bar{\mathbf{G}}_k \mathbf{q}_k^{(m)(i-1)} \quad \text{s.t.} \quad \bar{\mathbf{G}}_k = \begin{bmatrix} \mathbf{I}_M \\ -\tilde{\mathbf{G}}_k \end{bmatrix} \tag{2.70}$$

The weighted noisy covariance matrix $\bar{\mathbf{R}}_{\bar{y}\lambda,k}$ of the momentary frame and the past frames corresponding to the late reverberation can also be factorized as

$$\bar{\mathbf{R}}_{\bar{y}\lambda,k} = \begin{bmatrix} \mathbf{R}_{y\lambda,k} & \mathbf{P}_{y\lambda,k}^H \\ \mathbf{P}_{y\lambda,k} & \tilde{\mathbf{R}}_{\tilde{y}\lambda,k} \end{bmatrix} \tag{2.71}$$

into the weighted covariance matrix $\mathbf{R}_{y,k}$ of the momentary frame, the weighted covariance matrix $\tilde{\mathbf{R}}_{\tilde{y}\lambda,k}$ of the past frames corresponding to the late reverberation and the weighted cross-covariance matrix $\mathbf{P}_{y\lambda,k}$ of the past frames with the momentary frame. Using this in eq. (2.69) yields

$$\tilde{\mathbf{G}}_k^{(i)} = \underset{\tilde{\mathbf{G}}_k}{\operatorname{argmin}}\, \mathbf{q}_k^{(m)(i-1)H} \begin{bmatrix} \mathbf{I}_M \\ -\tilde{\mathbf{G}}_k \end{bmatrix}^H \begin{bmatrix} \mathbf{R}^{(i)}_{y\lambda,k} & \mathbf{P}^{(i)H}_{y\lambda,k} \\ \mathbf{P}^{(i)}_{y\lambda,k} & \tilde{\mathbf{R}}^{(i)}_{\tilde{y}\lambda,k} \end{bmatrix} \begin{bmatrix} \mathbf{I}_M \\ -\tilde{\mathbf{G}}_k \end{bmatrix} \mathbf{q}_k^{(m)(i-1)} \tag{2.72}$$

The solution of this optimization problem is equivalent to the conventional WPE dereverberation filter update given by eq. (2.31).

2.5.2.1.2 Estimation of Beamforming Filter $\mathbf{q}_k^{(m)}$ Using the dereverberation matrix $\bar{\mathbf{G}}_k^{(i)}$ estimated above on the noisy covariance matrix $\bar{\mathbf{R}}^{(i)}_{\bar{y}\lambda,k}$ provides the dereverberated signal $\breve{\mathbf{D}}_k^{(i)}$ with its weighted covariance matrix $\mathbf{R}^{(i)}_{\breve{d}\lambda,k}$ as

$$\breve{\mathbf{D}}_k^{(i)} = \bar{\mathbf{G}}_k^{(i)H} \bar{\mathbf{Y}}_k \quad \text{and} \quad \mathbf{R}^{(i)}_{\breve{d}\lambda,k} = \bar{\mathbf{G}}_k^{(i)H} \bar{\mathbf{R}}^{(i)}_{\bar{y}\lambda,k} \bar{\mathbf{G}}_k^{(i)} = \frac{1}{T} \breve{\mathbf{D}}_k^{(i)} \left(\mathbf{\Lambda}_k^{(i)}\right)^{-1} \breve{\mathbf{D}}_k^{(i)H} \tag{2.73}$$

The cost function to be optimized with respect to the beamforming filter $\mathbf{q}_k^{(m)}$ is then given by

$$\mathbf{q}_k^{(m)(i)} = \underset{\mathbf{q}_k^{(m)}}{\operatorname{argmin}} \left[\mathbf{q}_k^{(m)H} \mathbf{R}^{(i)}_{\breve{d}\lambda,k} \mathbf{q}_k^{(m)} \quad \text{s.t.} \quad \mathbf{q}_k^{(m)H} \tilde{\mathbf{v}}_k^{(m)} = 1 \right] \tag{2.74}$$

The constraint is rewritten due to the identity matrix in the M first rows of $\bar{\mathbf{G}}_k$ and the fact that only the first M entries of $\bar{\mathbf{v}}_k$ are nonzero. The solution to this optimization problem is equivalent to the MPDR solution in eq. (2.8) given by

$$\mathbf{q}_k^{(m)(i)} = \frac{\left(\mathbf{R}^{(i)}_{\breve{d}\lambda,k}\right)^{-1} \tilde{\mathbf{v}}_k^{(m)}}{\tilde{\mathbf{v}}_k^{(m)H} \left(\mathbf{R}^{(i)}_{\breve{d}\lambda,k}\right)^{-1} \tilde{\mathbf{v}}_k^{(m)}} \tag{2.75}$$

However due to the weights inside $\mathbf{R}^{(i)}_{\breve{d}\lambda,k}$ this is rather a wMPDR. The possibility of accessing the dereverberated signal is an advantage of the factorized algorithm. It enables e.g. new estimation of the SPP, the noise covariance matrix $\mathbf{R}_{n,k}$ and the RTF by CW in between the WPE stage and the wMPDR stage in each iteration of the optimization process. However the unified WPD algorithm has the advantage of fewer calculations per iteration, i.e. less computing cost per iteration. The complete workflow of the unified and the factorized version of MISO-WPD are presented in fig. 3.1 and algorithm 5 respectively, including an additional shape parameter p is as proposed in section 3.1.

Chapter 3

Proposed ℓ_p-Norm Reformulation for WPD

Similarly to the ℓ_p-norm reformulation of MISO-WPE in section 2.4.2.4 also the WPD algorithm can by reformulated using the ℓ_p-norm as its cost function. However in order to achieve denoising with an additional beamforming structure the MPDR constraint is added to the cost function. Furthermore in section 3.2 the ℓ_p-norm reformulation of WPD is modified using the mixed norm $\ell_{\mathbf{\Phi};2,p}$, which enables the derivation of a MIMO-WPD algorithm.

3.1 MISO-WPD Reformulation with ℓ_p-Norm

The cost function of MISO-WPD given in eq. (2.61) can be reformulated with the ℓ_p-norm as

$$\mathcal{C}\left(\mathbf{z}_k^{(m)}\right) = \left\|\mathbf{z}_k^{(m)}\right\|_p^p \quad \text{s.t.} \quad \bar{\mathbf{h}}_k^{(m)H}\bar{\mathbf{v}}_k^{(m)} = 1 \tag{3.1}$$

whereby p is the shape parameter of the sparsity promoting cost function, which is $p = 0$ in the conventional case, and $\bar{\mathbf{v}}_k^{(m)}$ is the zero padded RTF vector. The optimal filter vector $\bar{\mathbf{h}}_k^{(m)}$ is obtained by the constrained minimization of the cost function as

$$\begin{aligned} \bar{\mathbf{h}}_k^{(m)\mathrm{WPD}} &= \underset{\bar{\mathbf{h}}_k^{(m)}}{\operatorname{argmin}}\, \mathcal{C}\left(\mathbf{z}_k^{(m)}\right) \quad \text{s.t.} \quad \bar{\mathbf{h}}_k^{(m)H}\bar{\mathbf{v}}_k^{(m)} = 1 \\ &= \underset{\bar{\mathbf{h}}_k^{(m)}}{\operatorname{argmin}} \left\|\mathbf{z}_k^{(m)}\right\|_p^p \quad \text{s.t.} \quad \bar{\mathbf{h}}_k^{(m)H}\bar{\mathbf{v}}_k^{(m)} = 1 \end{aligned} \tag{3.2}$$

Analogously to section 2.4.2.4 the ℓ_p-norm optimization problem in eq. (3.2) is approached with a series of weighted ℓ_2-norm subproblems like in eq. (2.48), which

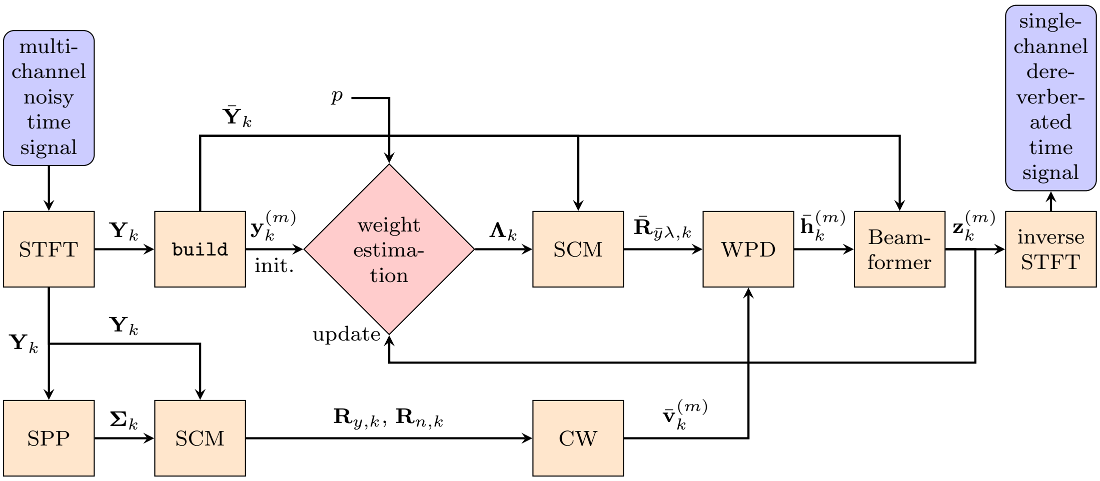

Figure 3.1: Flow chart of unified MISO-WPD.

are stated as

$$\begin{aligned} \bar{\mathbf{h}}_k^{(m)(i)} &= \underset{\bar{\mathbf{h}}_k^{(m)}}{\operatorname{argmin}}\, \mathbf{z}_k^{(m)} \left(\mathbf{\Lambda}_k^{(i)}\right)^{-1} \mathbf{z}_k^{(m)H} \quad \text{s.t.} \quad \bar{\mathbf{h}}_k^{(m)H}\bar{\mathbf{v}}_k^{(m)} = 1 \\ &= \underset{\bar{\mathbf{h}}_k^{(m)}}{\operatorname{argmin}}\, \bar{\mathbf{h}}_k^{(m)H}\bar{\mathbf{Y}}_k \left(\mathbf{\Lambda}_k^{(i)}\right)^{-1} \bar{\mathbf{Y}}_k^H\bar{\mathbf{h}}_k^{(m)} \quad \text{s.t.} \quad \bar{\mathbf{h}}_k^{(m)H}\bar{\mathbf{v}}_k^{(m)} = 1 \\ &= \underset{\bar{\mathbf{h}}_k^{(m)}}{\operatorname{argmin}}\, \bar{\mathbf{h}}_k^{(m)H}\bar{\mathbf{R}}_{\bar{y}\lambda,k}^{(i)}\bar{\mathbf{h}}_k^{(m)} \quad \text{s.t.} \quad \bar{\mathbf{h}}_k^{(m)H}\bar{\mathbf{v}}_k^{(m)} = 1 \end{aligned} \tag{3.3}$$

The solution to each of this subproblems is equivalent to the filter update of the conventional WPD algorithm in eq. (2.62) given by

$$\boxed{\bar{\mathbf{h}}_k^{(m)(i)} = \frac{\bar{\mathbf{R}}_{\bar{y}\lambda,k}^{(i)-1}\bar{\mathbf{v}}_k^{(m)}}{\bar{\mathbf{v}}_k^{(m)H}\bar{\mathbf{R}}_{\bar{y}\lambda,k}^{(i)-1}\bar{\mathbf{v}}_k^{(m)}}} \tag{3.4}$$

This optimal filter of the subproblem can now be used for the update of the weights $\lambda_{k,t}$ in the same way as in eq. (2.45), which results in

$$\boxed{\lambda_{k,t}^{(i)} = \left|z_{k,t}^{(m)(i-1)} + \varepsilon\right|^{2-p}} \tag{3.5}$$

whereby it is to notice that the WPD algorithm proposed by [29] is extended with a CGG sparse prior in the LH function [17], which leads to the subtraction of the shape parameter p in the exponent of the weight update. The complete workflow of the unified and factorized version (analogously to section 2.5.2) of MISO-WPD are described by fig. 3.1 and algorithm 5 respectively.

Algorithm 5: Factorized MISO-WPD (batch)

input : batch-matrix of multi-channel noisy microphone signal $\mathbf{Y}_k\ \forall\ k$

parameters: reference channel m, frequency dependent prediction delay τ_k, frequency dependent prediction filter length L_k, shape parameter p, regularization parameter ε, maximal number of iterations of the alternating optimization I_{max}, convergence tolerance η_c

functions : constructing convolutional signal matrix `build` $(\bullet)$, constructing diagonal matrix from vector `diagMat` $(\bullet)$, speech presence probability `spp` $(\bullet)$, trace of a matrix `trace` $(\bullet)$, covariance whitening `cw` $(\bullet)$

output : batch-vector of single-channel beamformed signal $\mathbf{z}_k^{(m)}\ \forall\ k$

```
foreach $k \in \{1, 2, \ldots, K\}$ do                                  // process each frequency subband $k$ individually
    $\tilde{\mathbf{Y}}_k = \texttt{build}\left(\mathbf{Y}_k; \tau_k, L_k\right)$      // builds convolutional signal matrix with past frames
    $\mathbf{\Lambda}_k = \texttt{diagMat}\left(\left|\mathbf{y}_k^{(m)}\right|^2 + \varepsilon\right)^{1-\frac{p}{2}}$      // initialize weights using the the noisy signal
    $\mathbf{z}_{k,\text{old}}^{(m)} = \mathbf{y}_k^{(m)}$                   // initialize $\mathbf{z}_{k,\text{old}}^{(m)}$ with the noisy signal
    for $i \leftarrow 1$ to $I_{max}$ do                                  // iterations of alternating optimization
        $\tilde{\mathbf{R}}_{\tilde{y}\lambda,k} = \frac{1}{T}\tilde{\mathbf{Y}}_k\mathbf{\Lambda}_k^{-1}\tilde{\mathbf{Y}}_k^H$      // estimate weighted noisy covariance matrix by SCM
        $\mathbf{P}_{y\lambda,k} = \frac{1}{T}\tilde{\mathbf{Y}}_k\mathbf{\Lambda}_k^{-1}\mathbf{Y}_k^H$      // estimate weighted noisy cross-covariance matrix by SCM
        $\tilde{\mathbf{G}}_k = \tilde{\mathbf{R}}_{\tilde{y}\lambda,k}^{-1}\mathbf{P}_{y\lambda,k}$      // estimate reverberation filter matrix
        $\breve{\mathbf{D}}_k = \mathbf{Y}_k - \tilde{\mathbf{G}}_k^H\tilde{\mathbf{Y}}_k$      // dereverberation by subtracting estimated reverberation
        $\mathbf{R}_{\breve{d}\lambda,k} = \frac{1}{T}\breve{\mathbf{D}}_k\mathbf{\Lambda}_k^{-1}\breve{\mathbf{D}}_k^H$      // estimate weighted dereverberated covariance matrix by SCM
        $\mathbf{\Sigma}_k = \texttt{spp}\left(\breve{\mathbf{D}}_k\right)$      // estimate SPP of dereverberated signal
        $\mathbf{R}_{n,k} = \frac{\breve{\mathbf{D}}_k(\mathbf{I}_{\mathbb{N}}-\mathbf{\Sigma}_k)\breve{\mathbf{D}}_k^H}{\texttt{trace}(\mathbf{I}_{\mathbb{N}}-\mathbf{\Sigma}_k)}$      // estimate noise covariance matrix by SCM using the SPP
        $\tilde{\mathbf{v}}_k^{(m)} = \texttt{cw}\left(\breve{\mathbf{D}}_k, \mathbf{R}_{n,k}, m\right)$      // estimating RTF-vector by CW (algorithm 2)
        $\mathbf{q}_k^{(m)} = \frac{\mathbf{R}_{\breve{d}\lambda,k}^{-1}\tilde{\mathbf{v}}_k^{(m)}}{\tilde{\mathbf{v}}_k^{(m)\,H}\mathbf{R}_{\breve{d}\lambda,k}^{-1}\tilde{\mathbf{v}}_k^{(m)}}$      // estimate wMPDR beamforming vector
        $\mathbf{z}_k^{(m)} = \mathbf{q}_k^{(m)\,H}\breve{\mathbf{D}}_k$      // estimate beamformed signal
        if $\frac{\left\|\mathbf{z}_k^{(m)} - \mathbf{z}_{k,\text{old}}^{(m)}\right\|_2}{\left\|\mathbf{z}_{k,\text{old}}^{(m)}\right\|_2} < \eta_c$ then      // convergence criterion $\eta$
            break      // breaks the for loop of the alternating optimization
        else
            $\mathbf{\Lambda}_k = \texttt{diagMat}\left(\left|\mathbf{z}_k^{(m)}\right|^2 + \varepsilon\right)^{1-\frac{p}{2}}$      // update weights using the beamformed signal
            $\mathbf{z}_{k,\text{old}}^{(m)} = \mathbf{z}_k^{(m)}$      // store output for convergence criterion of next iteration
```

3.2 Proposed MIMO-WPD beamforming

The first idea of the following proposed MIMO-WPD derivations was to provide a multi-channel beamformed signal $\mathbf{z}_{k,t}$ in order to use multiple channels for post-processing. However we show that the MIMO formulation mainly modifies the update of the weights $\lambda_{k,t}$. The underlying signal model is described in section 2.1 and the MIMO convolutional beamformer $\bar{\mathbf{H}}_k = \bar{\mathbf{G}}_k \mathbf{Q}_k = \left[\bar{\mathbf{h}}_k^{(1)}, \bar{\mathbf{h}}_k^{(2)}, \ldots, \bar{\mathbf{h}}_k^{(M)}\right] \in \mathbb{C}^{M(L_k-\tau_k+1)\times M}$ should perform joint dereverberation and denoising according to

$$\mathbf{z}_{k,t} = \bar{\mathbf{H}}_k^H \bar{\mathbf{y}}_{k,t} = \mathbf{Q}_k^H \bar{\mathbf{G}}_k^H \bar{\mathbf{y}}_{k,t} \tag{3.6}$$

whereby $\mathbf{Q}_k = \left[\mathbf{q}_k^{(1)}, \mathbf{q}_k^{(2)}, \ldots, \mathbf{q}_k^{(M)}\right] \in \mathbb{C}^{M\times M}$ is a MIMO-wMPDR beamformer matrix. Writing this filter equation in its batch formulation gives

$$\mathbf{Z}_k = \bar{\mathbf{H}}_k^H \bar{\mathbf{Y}}_k = \mathbf{Q}_k^H \bar{\mathbf{G}}_k^H \bar{\mathbf{Y}}_k \tag{3.7}$$

3.2.1 Filter Optimization

In order to find the optimal convolutional filter matrix I here propose the following MIMO extension of the cost function in eq. (3.2) analogously to eq. (2.52) as

$$\begin{aligned} \bar{\mathbf{H}}_k^{\mathrm{WPD}} &= \underset{\bar{\mathbf{H}}_k}{\operatorname{argmin}}\, \mathcal{C}\left(\mathbf{Z}_k\right) \quad \text{s.t.} \quad \bar{\mathbf{H}}_k^H \bar{\mathbf{v}}_k = \mathbf{v}_k \\ &= \underset{\bar{\mathbf{H}}_k}{\operatorname{argmin}} \|\mathbf{Z}_k\|_{\mathbf{\Phi};2,p}^p \quad \text{s.t.} \quad \bar{\mathbf{H}}_k^H \bar{\mathbf{v}}_k = \mathbf{v}_k \end{aligned} \tag{3.8}$$

The extension of the cost function compared to MIMO-WPE is given by the additional constraints for every channel. Here also the extension compared to

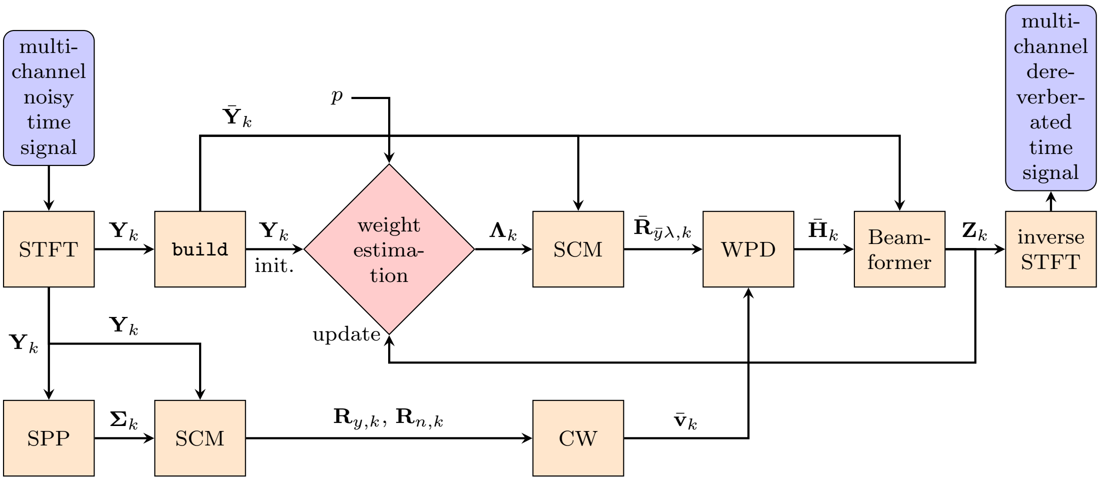

Figure 3.2: Flow chart of unified MIMO-WPD.

MISO-WPD can be noticed, since its cost function considered only the reference channel m. Important to notice here is that the multiple constraints of MIMO-WPD are described using the non-normalized mATF $\mathbf{v}_k$ and the corresponding zero-padded vector

$$\bar{\mathbf{v}}_k = \left[\mathbf{v}_k,\ \underbrace{0, 0, \ldots, 0}_{M(L_k-\tau_k)\text{ zeros}}\right]^T \in \mathbb{C}^{M(L_k-\tau_k+1)} \tag{3.9}$$

which can be obtained e.g. by CW (see eq. (2.19)). Like before this ℓ_p-norm optimization problem is approached with a series of weighted ℓ_2-norm subproblems like in eq. (3.3), which are stated as

$$\begin{aligned}
\bar{\mathbf{H}}_k^{(i)} &= \underset{\bar{\mathbf{H}}_k}{\operatorname{argmin}}\, \texttt{trace}\left(\mathbf{Z}_k\left(\mathbf{\Lambda}_k^{(i)}\right)^{-1}\mathbf{Z}_k^H\mathbf{\Phi}^{-1}\right) \quad \text{s.t.} \quad \bar{\mathbf{H}}_k^H\bar{\mathbf{v}}_k = \mathbf{v}_k \\
&= \underset{\bar{\mathbf{H}}_k}{\operatorname{argmin}}\, \texttt{trace}\left(\bar{\mathbf{H}}_k^H\mathbf{Y}_k\left(\mathbf{\Lambda}_k^{(i)}\right)^{-1}\mathbf{Y}_k^H\bar{\mathbf{H}}_k\mathbf{\Phi}^{-1}\right) \quad \text{s.t.} \quad \bar{\mathbf{H}}_k^H\bar{\mathbf{v}}_k = \mathbf{v}_k \\
&= \underset{\bar{\mathbf{H}}_k}{\operatorname{argmin}}\, \texttt{trace}\left(\bar{\mathbf{H}}_k^H\bar{\mathbf{R}}_{\bar{y}\lambda,k}^{(i)}\bar{\mathbf{H}}_k\mathbf{\Phi}^{-1}\right) \quad \text{s.t.} \quad \bar{\mathbf{H}}_k^H\bar{\mathbf{v}}_k = \mathbf{v}_k \\
&= \underset{\bar{\mathbf{H}}_k}{\operatorname{argmin}}\, \texttt{trace}\left(\bar{\mathbf{H}}_k\mathbf{\Phi}^{-1}\bar{\mathbf{H}}_k^H\bar{\mathbf{R}}_{\bar{y}\lambda,k}^{(i)}\right) \quad \text{s.t.} \quad \bar{\mathbf{v}}_k^H\bar{\mathbf{H}}_k = \mathbf{v}_k^H
\end{aligned} \tag{3.10}$$

whereby the following identity of traces is used: $\texttt{trace}(ABC) = \texttt{trace}(CAB) = \texttt{trace}(BCA)$. The solution to this problem can be obtained by using the method of Lagrange multipliers. The Lagrangian function $\mathfrak{L}$ with the auxiliary parameter vector $\boldsymbol{\alpha} \in \mathbb{C}^M$ for the constraint is to be optimized and can be formulated as (iteration index i omitted)

$$\mathfrak{L}\left(\bar{\mathbf{H}}_k, \boldsymbol{\alpha}\right) = \texttt{trace}\left(\bar{\mathbf{H}}_k\mathbf{\Phi}^{-1}\bar{\mathbf{H}}_k^H\bar{\mathbf{R}}_{\bar{y}\lambda,k}\right) + \left(\bar{\mathbf{v}}_k^H\bar{\mathbf{H}}_k - \mathbf{v}_k^H\right)\boldsymbol{\alpha} \tag{3.11}$$

The gradient of this Lagrangian function $\mathfrak{L}$ in respect to the convolutional filter matrix $\bar{\mathbf{H}}_k$ and the auxiliary parameter vector $\boldsymbol{\alpha}$ is set to zero, in order to find the optimum of the cost function. The following identity, shown in equation 233 from [35], is used to reformulate the complex gradient matrix

$$\nabla_{\bar{\mathbf{H}}_k}\mathfrak{L}\left(\bar{\mathbf{H}}_k, \boldsymbol{\alpha}\right) = 2\frac{\partial\mathfrak{L}\left(\bar{\mathbf{H}}_k, \boldsymbol{\alpha}\right)}{\partial\bar{\mathbf{H}}_k^*} \tag{3.12}$$

Inserting the Lagrangian function leads to

$$\nabla_{\bar{\mathbf{H}}_k}\mathfrak{L}\left(\bar{\mathbf{H}}_k,\boldsymbol{\alpha}\right)=\underbrace{2\frac{\partial}{\partial\bar{\mathbf{H}}_k^*}\mathtt{trace}\left(\bar{\mathbf{H}}_k\boldsymbol{\Phi}^{-1}\bar{\mathbf{H}}_k^H\bar{\mathbf{R}}_{\bar{y}\lambda,k}\right)}_{\text{A}}+2\frac{\partial}{\partial\bar{\mathbf{H}}_k^*}\left(\bar{\mathbf{v}}_k^H\bar{\mathbf{H}}_k-\mathbf{v}_k^H\right)\boldsymbol{\alpha} \tag{3.13}$$

whereby the derivative of the part A is given by table IV of [36] as

$$\frac{\partial}{\partial\bar{\mathbf{H}}_k^*}\mathtt{trace}\left(\bar{\mathbf{H}}_k\boldsymbol{\Phi}^{-1}\bar{\mathbf{H}}_k^H\bar{\mathbf{R}}_{\bar{y}\lambda,k}\right)=\bar{\mathbf{R}}_{\bar{y}\lambda,k}\bar{\mathbf{H}}_k\boldsymbol{\Phi}^{-1} \tag{3.14}$$

so that the complete gradient can be formulated as

$$\begin{aligned}&\nabla_{\bar{\mathbf{H}}_k}\mathfrak{L}\left(\bar{\mathbf{H}}_k,\boldsymbol{\alpha}\right)=2\left(\bar{\mathbf{R}}_{\bar{y}\lambda,k}\bar{\mathbf{H}}_k\boldsymbol{\Phi}^{-1}+\bar{\mathbf{v}}_k\boldsymbol{\alpha}^H\right)\overset{!}{=}\mathbf{0}\\&\Leftrightarrow\bar{\mathbf{R}}_{\bar{y}\lambda,k}\bar{\mathbf{H}}_k\boldsymbol{\Phi}^{-1}\overset{!}{=}-\bar{\mathbf{v}}_k\boldsymbol{\alpha}^H\\&\Leftrightarrow\bar{\mathbf{H}}_k\overset{!}{=}-\bar{\mathbf{R}}_{\bar{y}\lambda,k}^{-1}\bar{\mathbf{v}}_k\boldsymbol{\alpha}^H\boldsymbol{\Phi}\end{aligned} \tag{3.15}$$

whereby $\bar{\mathbf{R}}_{\bar{y}\lambda,k}$ is assumed to be invertible. The gradient in respect to the auxiliary parameter vector $\boldsymbol{\alpha}$ is given by the constraint itself as

$$\begin{aligned}&\nabla_{\boldsymbol{\alpha}}\mathfrak{L}\left(\bar{\mathbf{H}}_k,\boldsymbol{\alpha}\right)=\bar{\mathbf{v}}_k^H\bar{\mathbf{H}}_k-\mathbf{v}_k^H\overset{!}{=}\mathbf{0}\\&\Leftrightarrow\bar{\mathbf{v}}_k^H\bar{\mathbf{H}}_k\overset{!}{=}\mathbf{v}_k^H\\&\Leftrightarrow-\bar{\mathbf{v}}_k^H\bar{\mathbf{R}}_{\bar{y}\lambda,k}^{-1}\bar{\mathbf{v}}_k\boldsymbol{\alpha}^H\boldsymbol{\Phi}\overset{!}{=}\mathbf{v}_k^H\\&\Leftrightarrow\boldsymbol{\alpha}^H\overset{!}{=}\frac{-\mathbf{v}_k^H\boldsymbol{\Phi}^{-1}}{\bar{\mathbf{v}}_k^H\bar{\mathbf{R}}_{\bar{y}\lambda,k}^{-1}\bar{\mathbf{v}}_k}\end{aligned} \tag{3.16}$$

where the result of eq. (3.15) is inserted in the second step. The solution can be obtained by inserting $\boldsymbol{\alpha}^H$ from eq. (3.16) into eq. (3.15)

$$\boxed{\bar{\mathbf{H}}_k^{(i)}=\frac{\bar{\mathbf{R}}_{\bar{y}\lambda,k}^{(i)-1}\bar{\mathbf{v}}_k\mathbf{v}_k^H}{\bar{\mathbf{v}}_k{}^H\bar{\mathbf{R}}_{\bar{y}\lambda,k}^{(i)-1}\bar{\mathbf{v}}_k}} \tag{3.17}$$

The update of the weights $\lambda_{k,t}$ is equivalent to the weight update of MIMO-WPE in eq. (2.54) and can also be regularized by a small positive constant ε in order to avoid division by zero.

$$\boxed{\lambda_{k,t}^{(i)}=\left\|\mathbf{z}_{k,t}^{(i-1)}+\varepsilon\right\|_{\boldsymbol{\Phi};2}^{2-p}} \tag{3.18}$$

The complete workflow of the unified MIMO-WPD is presented in fig. 3.2.

3.2.2 Factorized MIMO-WPD

Similar to section 2.5.2 also a factorized formulation of MIMO-WPD is possible. Hereby the convolutional filter matrix $\bar{\mathbf{H}}_k$ factorizes into the dereverberation filter matrix $\bar{\mathbf{G}}_k$ and the beamforming filter matrix $\mathbf{Q}_k$ as

$$\bar{\mathbf{H}}_k = \bar{\mathbf{G}}_k \mathbf{Q}_k \tag{3.19}$$

The estimation of $\bar{\mathbf{G}}_k$ as the first step is completely equivalent to its MISO version described in paragraph 2.5.2.1.1. The cost function for the beamforming filter matrix $\mathbf{Q}_k$ can be formulated as

$$\mathbf{Q}_k^{(i)} = \underset{\mathbf{Q}_k}{\operatorname{argmin}}\, \mathbf{Q}_k^H \mathbf{R}_{\breve{d}\lambda,k}^{(i)} \mathbf{Q}_k \quad \text{s.t.} \quad \mathbf{Q}_k^H \mathbf{v}_k = \mathbf{v}_k \tag{3.20}$$

It is to be noticed that here again the mATF instead of the RTF is used to define the distortionless constraint for each channel in a parallel manner. The solution to this optimization problem is similarly to eq. (3.17) given by

$$\mathbf{Q}_k^{(i)} = \frac{\mathbf{R}_{\breve{d}\lambda,k}^{(i)-1} \mathbf{v}_k \mathbf{v}_k^H}{\mathbf{v}_k{}^H \mathbf{R}_{\breve{d}\lambda,k}^{(i)-1} \mathbf{v}_k} \tag{3.21}$$

which can be referred to as a MIMO-wMPDR beamformer. The factorized solution has a general advantage over the unified solution: It is possible to get access to the dereverberated signal, which e.g. can offer new estimation of the SPP, the resulting noise covariance matrix $\mathbf{R}_{n,k}$ and the resulting RTF by CW in between the WPE stage and the wMPDR stage in each iteration of the optimization process. However the unified WPD algorithm has the advantage of fewer calculations per iteration, which means less computing power per iteration is required. The complete workflow of the factorized MIMO-WPD is described in algorithm 6.

Algorithm 6: Factorized MIMO-WPD (batch)

input : batch-matrix of multi-channel noisy microphone signal $\mathbf{Y}_k \ \forall \ k$

parameters: frequency dependent prediction delay τ_k, frequency dependent prediction filter length L_k, shape parameter p, regularization parameter ε, maximal number of iterations of the alternating optimization I_{max}, convergence tolerance η_c

functions : constructing convolutional signal matrix `build` $(\bullet)$, constructing diagonal matrix from vector `diagMat` $(\bullet)$, speech presence probability `spp` $(\bullet)$, trace of a matrix `trace` $(\bullet)$, covariance whitening `cw` $(\bullet)$

output : batch-matrix of multi-channel beamformed signal $\mathbf{Z}_k \ \forall \ k$

```
foreach $k \in \{1, 2, \ldots, K\}$ do    // process each frequency subband $k$ individually
    $\tilde{\mathbf{Y}}_k = \texttt{build}\left(\mathbf{Y}_k; \tau_k, L_k\right)$    // builds convolutional signal matrix with past frames
    foreach $t \in \{1, 2, \ldots, T\}$ do    // process each time frame $t$ individually
        $\lambda_{k,t} = \left(\|\mathbf{y}_{k,t}\|^2_{\mathbf{\Phi};2} + \varepsilon\right)^{1-\frac{p}{2}}$    // initialize weights by mixed norm of the noisy signal
    $\mathbf{\Lambda}_k = \texttt{diagMat}\left(\boldsymbol{\lambda}_k = [\lambda_{k,1}, \lambda_{k,2}, \ldots, \lambda_{k,T}]\right)$
    $\mathbf{Z}_{k,\text{old}} = \mathbf{Y}_k$    // initialize $\mathbf{Z}_{k,\text{old}}$ with the noisy signal
    for $i \leftarrow 1$ to $I_{max}$ do    // iterations of alternating optimization
        $\tilde{\mathbf{R}}_{\tilde{y}\lambda,k} = \frac{1}{T}\tilde{\mathbf{Y}}_k \mathbf{\Lambda}_k^{-1} \tilde{\mathbf{Y}}_k^H$    // estimate weighted noisy covariance matrix by SCM
        $\mathbf{P}_{y\lambda,k} = \frac{1}{T}\tilde{\mathbf{Y}}_k \mathbf{\Lambda}_k^{-1} \mathbf{Y}_k^H$    // estimate weighted noisy cross-covariance matrix by SCM
        $\tilde{\mathbf{G}}_k = \tilde{\mathbf{R}}_{\tilde{y}\lambda,k}^{-1} \mathbf{P}_{y\lambda,k}$    // estimate reverberation filter matrix
        $\breve{\mathbf{D}}_k = \mathbf{Y}_k - \tilde{\mathbf{G}}_k^H \tilde{\mathbf{Y}}_k$    // dereverberation by subtracting estimated reverberation
        $\mathbf{R}_{\breve{d}\lambda,k} = \frac{1}{T}\breve{\mathbf{D}}_k \mathbf{\Lambda}_k^{-1} \breve{\mathbf{D}}_k^H$    // estimate weighted dereverberated covariance matrix by SCM
        $\mathbf{\Sigma}_k = \texttt{spp}\left(\breve{\mathbf{D}}_k\right)$    // estimate SPP of dereverberated signal
        $\mathbf{R}_{n,k} = \frac{\breve{\mathbf{D}}_k (\mathbf{I}_{\mathbb{N}} - \mathbf{\Sigma}_k) \mathbf{\Lambda}_k^{-1} \breve{\mathbf{D}}_k^H}{\texttt{trace}(\mathbf{I}_{\mathbb{N}} - \mathbf{\Sigma}_k)}$    // estimate noise covariance matrix by SCM using the SPP
        $\mathbf{v}_k = \texttt{cw}\left(\breve{\mathbf{D}}_k, \mathbf{R}_{n,k}\right)$    // estimating mATF-vector by CW (algorithm 2)
        $\mathbf{Q}_k = \frac{\mathbf{R}_{\breve{d}\lambda,k}^{-1} \mathbf{v}_k \mathbf{v}_k^H}{\mathbf{v}_k^H \mathbf{R}_{\breve{d}\lambda,k}^{-1} \mathbf{v}_k}$    // estimate multi-channel wMPDR beamforming matrix
        $\mathbf{Z}_k = \mathbf{Q}_k^H \breve{\mathbf{D}}_k$    // perform beamforming on dereverberated signal
        if $\frac{\|\mathbf{Z}_k - \mathbf{Z}_{k,\text{old}}\|_{\text{Fro}}}{\|\mathbf{Z}_{k,\text{old}}\|_{\text{Fro}}} < \eta_c$ then    // relative convergence criterion $\eta$
            break    // breaks the for loop of the alternating optimization
        else
            foreach $t \in \{1, 2, \ldots, T\}$ do    // process each time frame $t$ individually
                $\lambda_{k,t} = \left(\|\mathbf{z}_{k,t}\|^2_{\mathbf{\Phi};2} + \varepsilon\right)^{1-\frac{p}{2}}$    // updating weights by mixed norm of $\mathbf{z}_{k,t}$
            $\mathbf{\Lambda}_k = \texttt{diagMat}\left(\boldsymbol{\lambda}_k = [\lambda_{k,1}, \lambda_{k,2}, \ldots, \lambda_{k,T}]\right)$
            $\mathbf{Z}_{k,\text{old}} = \mathbf{Z}_k$    // store output for convergence criterion of next iteration
```

3.2.3 MISO-WPD with MIMO Weight Update

An investigation of the proposed MIMO-WPD algorithm reveals that an equivalent MISO formulation of this algorithm is possible by only adjusting the update of the weights $\lambda_{k,t}$. The MISO-WPD formulation is given by

$$\begin{aligned} z_{k,t}^{(m)(i)} &= \bar{\mathbf{h}}_k^{(m)(i)H}\bar{\mathbf{y}}_{k,t} = \frac{\bar{\mathbf{v}}_k^{(m)H}\bar{\mathbf{R}}_{\bar{y}\lambda,k}^{(i)-1}}{\bar{\mathbf{v}}_k^{(m)H}\bar{\mathbf{R}}_{\bar{y}\lambda,k}^{(i)-1}\bar{\mathbf{v}}_k^{(m)}}\bar{\mathbf{y}}_{k,t} \\ &= \frac{\left(\frac{\bar{\mathbf{v}}_k}{v_k^{(m)}}\right)^H\bar{\mathbf{R}}_{\bar{y}\lambda,k}^{(i)-1}}{\left(\frac{\bar{\mathbf{v}}_k}{v_k^{(m)}}\right)^H\bar{\mathbf{R}}_{\bar{y}\lambda,k}^{(i)-1}\frac{\bar{\mathbf{v}}_k}{v_k^{(m)}}}\bar{\mathbf{y}}_{k,t} = v_k^{(m)}\underbrace{\frac{\bar{\mathbf{v}}_k{}^H\bar{\mathbf{R}}_{\bar{y}\lambda,k}^{(i)-1}}{\bar{\mathbf{v}}_k{}^H\bar{\mathbf{R}}_{\bar{y}\lambda,k}^{(i)-1}\bar{\mathbf{v}}_k}\bar{\mathbf{y}}_{k,t}}_{\text{reference}-\text{independent}} \end{aligned} \tag{3.22}$$

This final term has a reference-independent part, which is additionally multiplied with the mATF value corresponding to the reference channel m, to obtain the MISO-WPD beamformer. This provides the possibility of formulating a MIMO-WPD filter operation by stacking the STFT-coefficients of the beamformed signal and the mATF as

$$\mathbf{z}_{k,t}^{(i)} = \frac{\mathbf{v}_k\bar{\mathbf{v}}_k{}^H\bar{\mathbf{R}}_{\bar{y}\lambda,k}^{(i)-1}}{\bar{\mathbf{v}}_k{}^H\bar{\mathbf{R}}_{\bar{y}\lambda,k}^{(i)-1}\bar{\mathbf{v}}_k}\bar{\mathbf{y}}_{k,t} = \bar{\mathbf{H}}_k^{(i)H}\bar{\mathbf{y}}_{k,t} \tag{3.23}$$

Here it is proven that the proposed MIMO-WPD performs a MISO-WPD for each channel in a parallel manner. However this is only true for one iteration through the series of ℓ_2-norm subproblems, because it additionally modifies the weight update. Since $\mathbf{v}_k = v_k^{(m)}\tilde{\mathbf{v}}_k^{(m)}$ this can be reformulated as

$$\mathbf{z}_{k,t}^{(i)} = \frac{v_k^{(m)}\tilde{\mathbf{v}}_k^{(m)}\bar{\mathbf{v}}_k{}^H\bar{\mathbf{R}}_{\bar{y}\lambda,k}^{(i)-1}}{\bar{\mathbf{v}}_k{}^H\bar{\mathbf{R}}_{\bar{y}\lambda,k}^{(i)-1}\bar{\mathbf{v}}_k}\bar{\mathbf{y}}_{k,t} = \tilde{\mathbf{v}}_k^{(m)}z_{k,t}^{(m)(i)} \tag{3.24}$$

The algorithm can now be adjusted to a MISO version by shifting the RTF $\tilde{\mathbf{v}}_k^{(m)}$ from the MIMO-WPD filter operation in eq. (3.24) into the update of the weights $\lambda_{k,t}$. The MISO-WPD weight update is given by

$$\lambda_{k,t}^{(i)} = \left|z_{k,t}^{(m)(i-1)}\right|^{2-p} \tag{3.25}$$

The weight update of the MISO-WPD algorithm can be modified to be equivalent to MIMO-WPD by inserting the RTF $\tilde{\mathbf{v}}_k^{(m)}$ from eq. (3.24) into eq. (3.25) as

$$\boxed{\lambda_{k,t}^{(i)} = \left\|\mathbf{z}_{k,t}^{(i-1)}\right\|_{\mathbf{\Phi};2}^{2-p} = \left\|z_{k,t}^{(m)(i-1)}\tilde{\mathbf{v}}_k^{(m)}\right\|_{\mathbf{\Phi};2}^{2-p} = \left|z_{k,t}^{(m)(i-1)}\right|^{2-p}\left\|\tilde{\mathbf{v}}_k^{(m)}\right\|_{\mathbf{\Phi};2}^{2-p}} \tag{3.26}$$

An overview of the complete workflow of factorized MIMO equivalent MISO-WPD is presented in algorithm 7.

Algorithm 7: Factorized MIMO equivalent MISO-WPD (batch)

```
input      : batch-matrix of multi-channel noisy microphone signal $\mathbf{Y}_k \ \forall \ k$
parameters: reference channel $m$, frequency dependent prediction delay $\tau_k$,
             frequency dependent prediction filter length $L_k$, shape parameter $p$,
             regularization parameter $\varepsilon$, maximal number of iterations of the
             alternating optimization $I_{max}$, convergence tolerance $\eta_c$
functions  : constructing convolutional signal matrix build(•), constructing
             diagonal matrix from vector diagMat(•), speech presence probability
             spp(•), trace of a matrix trace(•), covariance whitening cw(•)
output     : batch-vector of single-channel beamformed signal $\mathbf{z}_k^{(m)} \ \forall \ k$

foreach $k \in \{1, 2, \dots, K\}$ do                    // process each frequency subband $k$ individually
    $\tilde{\mathbf{Y}}_k = \texttt{build}(\mathbf{Y}_k; \tau_k, L_k)$        // builds convolutional signal matrix with past frames
    foreach $t \in \{1, 2, \dots, T\}$ do                // process each time frame $t$ individually
        $\lambda_{k,t} = \left(\|\mathbf{y}_{k,t}\|^2_{\mathbf{\Phi};2} + \varepsilon\right)^{1-\frac{p}{2}}$     // initialize weights by mixed norm of the noisy signal
    $\mathbf{\Lambda}_k = \texttt{diagMat}(\boldsymbol{\lambda}_k = [\lambda_{k,1}, \lambda_{k,2}, \dots, \lambda_{k,T}])$
    $\mathbf{z}_{k,\text{old}}^{(m)} = \mathbf{y}_k^{(m)}$                      // initialize $\mathbf{z}_{k,\text{old}}^{(m)}$ with the noisy signal
    for $i \leftarrow 1$ to $I_{max}$ do                 // iterations of alternating optimization
        $\tilde{\mathbf{R}}_{\tilde{y}\lambda,k} = \frac{1}{T}\tilde{\mathbf{Y}}_k \mathbf{\Lambda}_k^{-1} \tilde{\mathbf{Y}}_k^H$      // estimate weighted noisy covariance matrix by SCM
        $\mathbf{P}_{y\lambda,k} = \frac{1}{T}\tilde{\mathbf{Y}}_k \mathbf{\Lambda}_k^{-1} \mathbf{Y}_k^H$      // estimate weighted noisy cross-covariance matrix by SCM
        $\tilde{\mathbf{G}}_k = \tilde{\mathbf{R}}_{\tilde{y}\lambda,k}^{-1} \mathbf{P}_{y\lambda,k}$           // estimate reverberation filter matrix
        $\breve{\mathbf{D}}_k = \mathbf{Y}_k - \tilde{\mathbf{G}}_k^H \tilde{\mathbf{Y}}_k$          // dereverberation by subtracting estimated reverberation
        $\mathbf{R}_{\breve{d}\lambda,k} = \frac{1}{T}\breve{\mathbf{D}}_k \mathbf{\Lambda}_k^{-1} \breve{\mathbf{D}}_k^H$     // estimate weighted dereverberated covariance matrix by SCM
        $\mathbf{\Sigma}_k = \texttt{spp}\left(\breve{\mathbf{D}}_k\right)$                // estimate SPP of dereverberated signal
        $\mathbf{R}_{n,k} = \frac{\breve{\mathbf{D}}_k(\mathbf{I}_{\mathbb{N}} - \mathbf{\Sigma}_k)\breve{\mathbf{D}}_k^H}{\texttt{trace}(\mathbf{I}_{\mathbb{N}} - \mathbf{\Sigma}_k)}$   // estimate noise covariance matrix by SCM using the SPP
        $\tilde{\mathbf{v}}_k^{(m)} = \texttt{cw}\left(\breve{\mathbf{D}}_k, \mathbf{R}_{n,k}, m\right)$        // estimating RTF-vector by CW (algorithm 2)
        $\mathbf{q}_k^{(m)} = \frac{\mathbf{R}_{\breve{d}\lambda,k}^{-1}\tilde{\mathbf{v}}_k^{(m)}}{\tilde{\mathbf{v}}_k^{(m)\,H}\mathbf{R}_{\breve{d}\lambda,k}^{-1}\tilde{\mathbf{v}}_k^{(m)}}$        // estimate wMPDR beamforming vector
        $\mathbf{z}_k^{(m)} = \mathbf{q}_k^{(m)\,H}\breve{\mathbf{D}}_k$                   // estimate beamformed signal
        if $\frac{\left\|\mathbf{z}_k^{(m)} - \mathbf{z}_{k,\text{old}}^{(m)}\right\|_2}{\left\|\mathbf{z}_{k,\text{old}}^{(m)}\right\|_2} < \eta_c$ then      // convergence criterion $\eta$
            break                            // breaks the for loop of the alternating optimization
        else
            $\mathbf{\Lambda}_k = \texttt{diagMat}\left(\left|\mathbf{z}_k^{(m)}\right|^2 \left\|\tilde{\mathbf{v}}_k^{(m)}\right\|^2_{\mathbf{\Phi};2} + \varepsilon\right)^{1-\frac{p}{2}}$   // update weights
            $\mathbf{z}_{k,\text{old}}^{(m)} = \mathbf{z}_k^{(m)}$      // store output for convergence criterion of next iteration
```

Chapter 4

Evaluation by Experiments

The main objective of the evaluation experiments is the comparison of the proposed MIMO-WPD algorithm with the conventional MISO-WPD algorithm. So each of the following experiments is performed using both algorithms. After introduction of the experimental environment in section 4.1 the following two experiments are performed.

- Evaluation of convergence performance (section 4.2)
- Influence of shape parameter p (section 4.3)

4.1 Experimental Environment

Here the main process flow of the experiments is further described. To this end, the dataset, the objective measures and the choice of parameters for the algorithms are introduced.

4.1.1 Dataset

To be able to directly compare results to the recent publications of WPD [1, 29, 30] the evaluation was performed on the Reverb Challenge dataset [37]. This dataset was built assuming a single stationary speaker uttering sentences within different rooms and it contains speech signals of real and simulated data recorded by a 1-channel, a 2-channel and an 8-channel circular microphone array. However in this work only the simulated 8-channel recordings of the development set are used for evaluation. This subset contains 1484 recordings with a total length of about three hours, whereby the sentences are uttered by 10 different speakers. These recordings are based on clean signals from the WSJCAM0 corpus

[38], which are convolved with measured RIRs and further degraded by additive stationary noise signals with a fixed SNR of about 20 dB. The RIRs correspond to three different rooms with reverberation times T_{60} of

- $\text{RIR}_1 \rightarrow T_{60} = 250\,\text{ms}$
- $\text{RIR}_2 \rightarrow T_{60} = 500\,\text{ms}$
- $\text{RIR}_3 \rightarrow T_{60} = 700\,\text{ms}$

For each room two scenarios with different microphone array to speaker distances are recorded:

- ”near” → 100 cm
- ”far” → 250 cm

which leads to six different multi-channel RIRs. For the first two experiments the Reverb Challenge dataset is used, whereby each utterance is convolved with one of the ”near” and the ”far” RIRs for one of the three rooms (chosen randomly). These simulated audio snippets are used for the experiments further described in section 4.2 and section 4.3.

4.1.2 Objective Measures of Speech Quality

Objective measures are utilized to enable proper comparison of the algorithms. Reference signals are needed for most of the objective measures, for which we used the clean signals. The following three objective measures are evaluated:

- **Perceptual evaluation of speech quality (PESQ)** [39]:
  An objective measure of the perceptual speech quality requiring a reference signal. Larger values indicate better speech quality.

- **Frequency weighted segmetal SNR (FWSSNR)** [40]:
  An objective measure which performs a frequency dependent weighting on the segmental SNR, for which a reference signal is necessary. Values are given in dB, whereby larger values indicate better speech quality.

- **Cepstral distance (CD)** [40]:
  An objective measure providing an estimate of the logarithmic distance between two magnitude spectra. Smaller values indicate that the output signal is closer to the clean reference signal, which is preferred.

4.1.3 Choice of Parameters

In order to perform the suggested experiments with the described algorithms a number of parameters has to be set. Parameters that were not tuned were set to standard values and kept constant across evaluations. Unless mentioned differently in a particular experiment the following values shown in table 4.1 are used for the evaluation experiments. The sampling frequency of the data is already fixed by the choice of the dataset. The STFT transform was chosen as a basic transformation from the time domain into a framed frequency domain. The specific parameters of the STFT transform are chosen according to [1], which is also used as a motivation to chose the same frequency dependent prediction delays and same frequency dependent prediction filter lengths. For the experiments an energy-based voice activity detection (VAD) of the noisy microphone signal is estimated, which is then used to determine the noise covariance matrix by SCM similar as in eq. (2.12). It replaced the SPP in the formulated algorithms, but was not updated in each iteration. However the noise is estimated only by the noisy frames before the first and after the last speech frame, which are detected by the VAD.

Table 4.1: Choice of fixed parameters for the WPD algorithms.

Parameter	**Symbol**	**Value**
sampling frequency		16 kHz
transform		STFT
frame length		1024 taps $\widehat{=}$ 64 ms
frame shift		256 taps $\widehat{=}$ 16 ms
window		Hamming
prediction delay	τ_k	4 frames $\widehat{=}$ 64 ms
prediction filter length (0 kHz - 0.8 kHz)	$L_{k=[0,51]}$	12 frames $\widehat{=}$ 192 ms
prediction filter length (0.8 kHz - 1.5 kHz)	$L_{k=[52,96]}$	10 frames $\widehat{=}$ 160 ms
prediction filter length (1.5 kHz - 8 kHz)	$L_{k=[97,512]}$	6 frames $\widehat{=}$ 96 ms
regularization constant	ε	1×10^{-8}
reference channel	m	1

4.2 Convergence Performance

In the first step the convergence performance of the MIMO and MISO-WPD algorithm was investigated. The following maximal numbers of iterations per frequency of the alternating optimization procedure were chosen:

$$I_{max} = \{1, 2, 3, 4, 5, 7, 9, 11, 13, 15, 18, 21, 24, 27, 30, 34, 38, 42, 46, 50, 55\} \quad (4.1)$$

The notches in the following boxplots indicate whether medians of two distributions are significantly different from each another. Figure 4.1 shows clearly that the median over all signals of the convergence criterion η is significantly smaller for the MIMO-WPD algorithm compared to MISO-WPD. This indicates that the MIMO version of the algorithm converges more quickly than the MISO version.

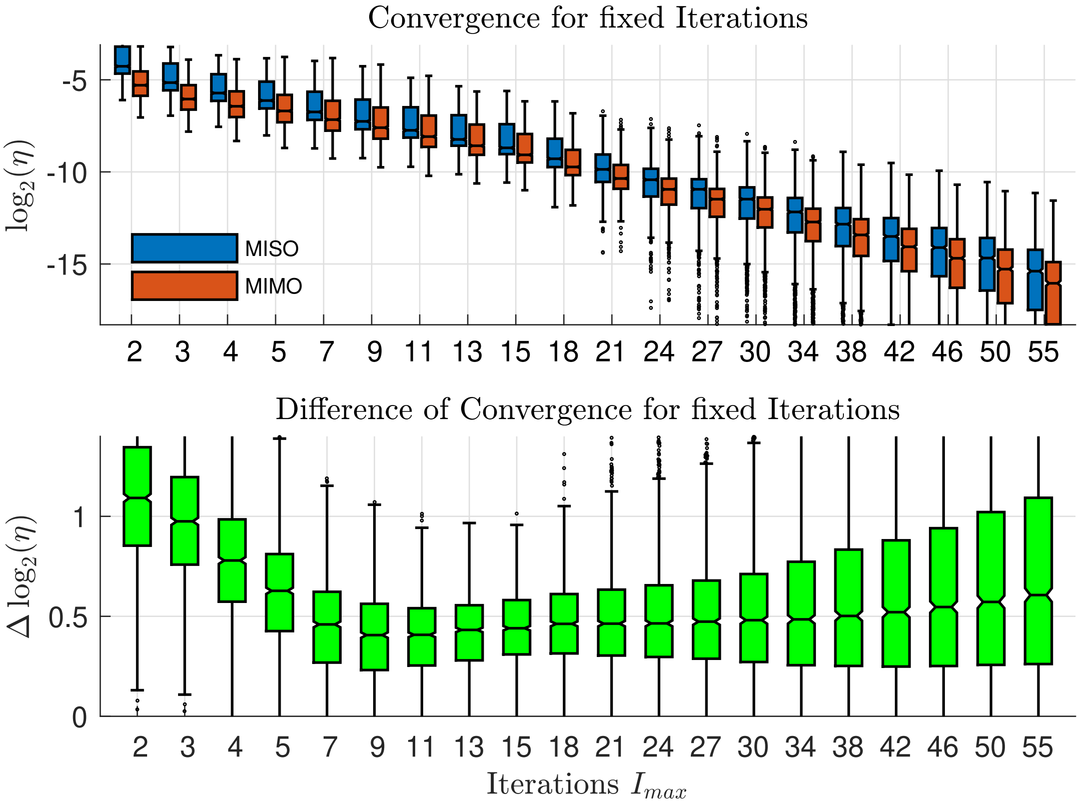

Figure 4.1: This boxplot shows the distributions of the base-2-logarithm of the convergence criterion η over the maximal iterations. The distributions contain the convergence per signal, which is defined as the geometric mean of the convergence criteria for each frequency subband. The upper plot shows a comparison between the MIMO-WPD algorithm and the MISO-WPD algorithm, whereas the lower plot shows the difference of the two. Hereby positive values indicate smaller convergence criteria η for MIMO-WPD.

In the second step the tolerance η_c for the convergence criterion η was fixed and the amount of iterations needed for convergence was determined. The following values for the tolerance η_c of the convergence criterion were chosen:

$$\eta_c = \frac{1}{2^b} \quad \text{with} \quad b = \{0, 1, 2, \dots, 20\} \tag{4.2}$$

Figure 4.2 shows that the median over all signals of the mean iterations until convergence for MIMO-WPD is significantly smaller for tolerances of $\eta_c < 2^{-3}$, compared to MISO-WPD. This indicates that the MIMO version converges with fewer iterations than its MISO equivalent.

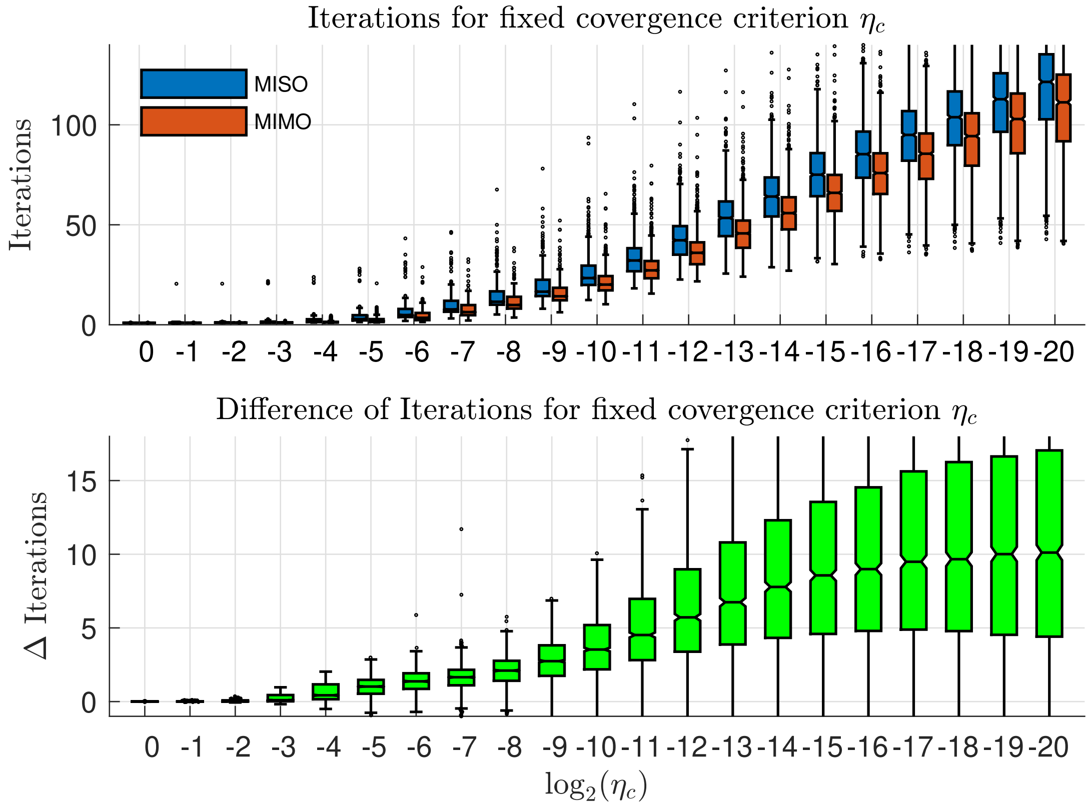

Figure 4.2: This boxplot shows the distributions of the iterations needed to converge for fixed tolerances η_c of the convergence criteria. The distributions contain the mean iteration per signal, which is defined as the arithmetic mean of the iterations for each frequency subband. The upper plot shows a comparison between the MIMO-WPD algorithm and the MISO-WPD algorithm, whereas the lower plot shows the difference of the two. Hereby positive values indicate fewer iterations for MIMO-WPD.

4.2.1 Convergence of Objective Speech Quality Measures

Since we aim at improving the speech quality, we are interested in the performance measured by the objective speech quality measures. Therefore the dependency of the objective speech quality measures described in section 4.1.2 over the maximal number of iterations I_{max} is presented in the following. Hereby fixing the maximal number of iterations I_{max} in contrast to fixing the tolerance η_c of the convergence criterion has the advantage, that the computing time is deterministic, i.e. predictable. Figure 4.3 reveals that the MIMO version of the WPD algorithm significantly outperforms the MISO version in terms of PESQ improvement, since the median after convergence is 0.07 larger. MIMO-WPD also converges faster with only 3 iterations compared to 7 iterations for the MISO version. The median of the PESQ improvement for MIMO-WPD after only one iteration is significantly larger than the median of PESQ improvement for MISO-WPD after convergence.

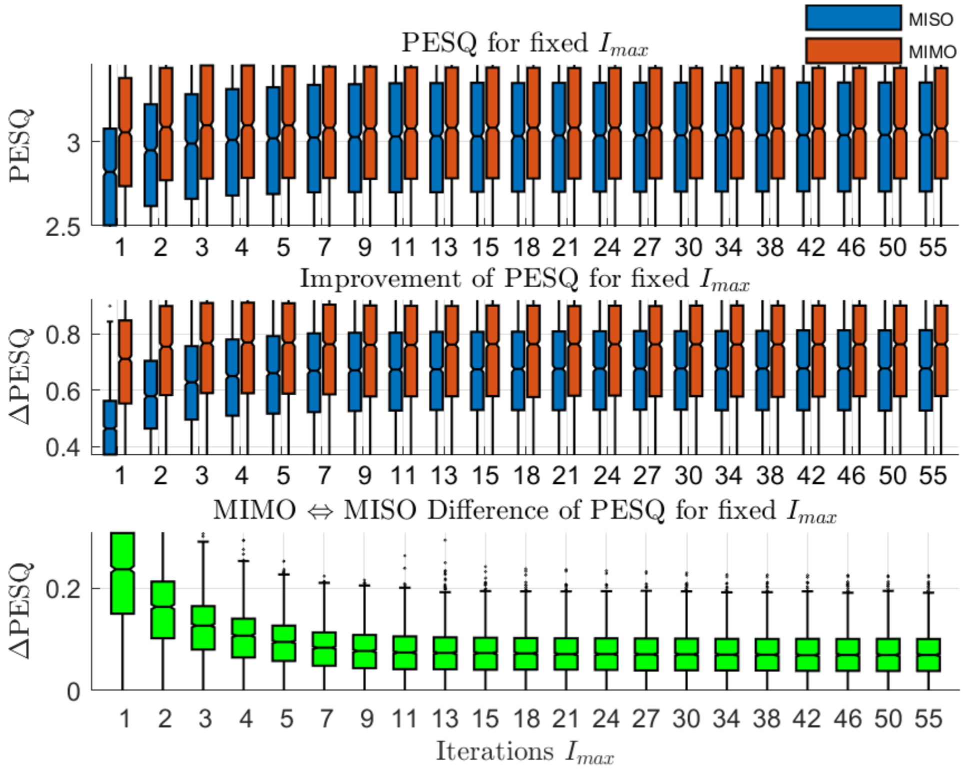

Figure 4.3: This boxplot shows the distributions of the PESQ of the beamformed signals and its improvements over the iterations I_{max}. The upper plot shows a comparison of the absolute PESQ between the MIMO-WPD and the MISO-WPD algorithm, the middle plot shows a comparison of the PESQ improvement compared to the noisy input signal between the MIMO-WPD and the MISO-WPD algorithm, whereas the lowest plot shows the difference of the two. In the lowest plot positive values indicate a larger PESQ improvement for MIMO-WPD.

Figure 4.4 reveals that MIMO-WPD also outperforms MISO-WPD in terms of FWSSNR after both versions of the WPD algorithm converged, with a 0.35 dB larger median of the FWSSNR improvement. Additionally MIMO-WPD reaches convergence of the FWSSNR after 3 iterations, whereas MISO-WPD needs 7 iterations. The median after convergence of FWSSNR improvement for MISO-WPD is not significantly larger than the median of FWSSNR improvement for MIMO-WPD after only one iteration.

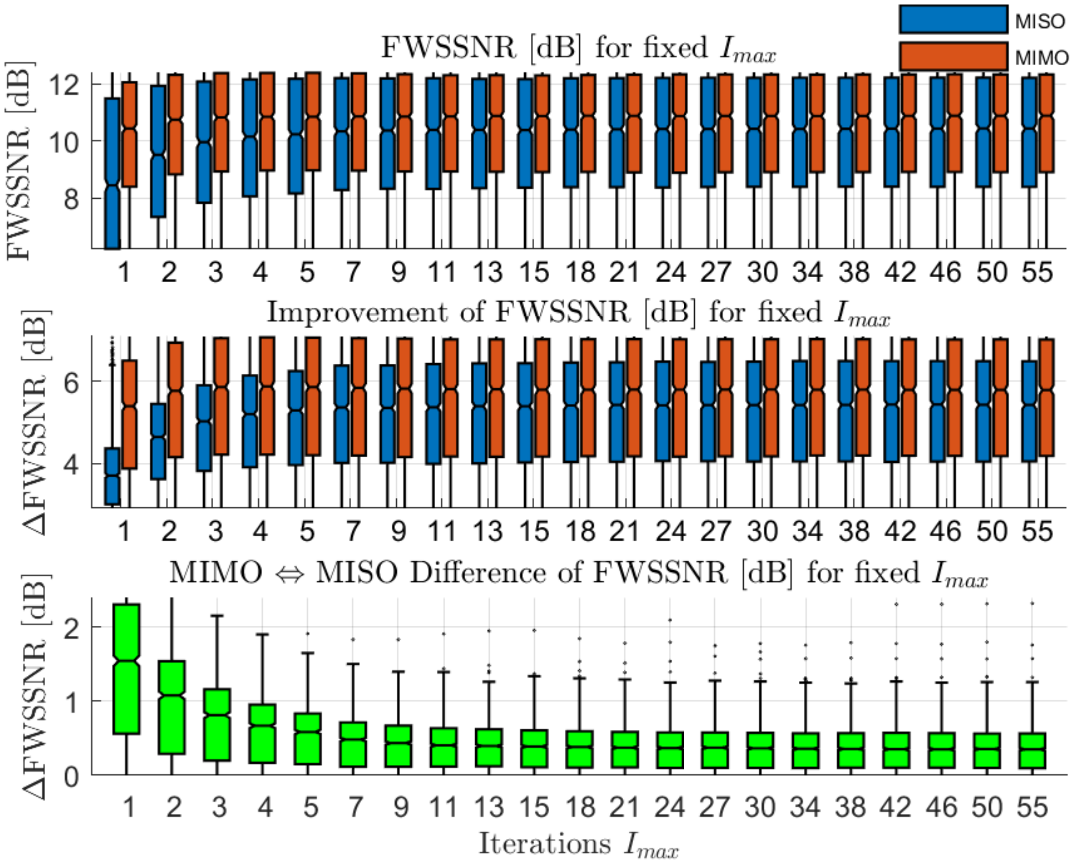

Figure 4.4: This boxplot shows the distributions of the FWSSNR of the beamformed signals and its improvements over the iterations I_{max}. The upper plot shows a comparison of the absolute FWSSNR between the MIMO-WPD and the MISO-WPD algorithm, the middle plot shows a comparison of the FWSSNR improvement compared to the noisy input signal between the MIMO-WPD and the MISO-WPD algorithm, whereas the lowest plot shows the difference of the two. In the lowest plot positive values indicate more FWSSNR improvement for MIMO-WPD.

Figure 4.5 shows that MIMO-WPD additionally outperforms MISO-WPD in terms of CD after both algorithms converged, with a 0.1 larger median of the CD improvement. For the CD improvement MIMO-WPD converges after 4 iterations, whereas the MISO version needs 9 iterations. The median of the CD improvement for MIMO-WPD after only one iteration is insignificantly larger than the median of CD improvement for MISO-WPD after convergence.

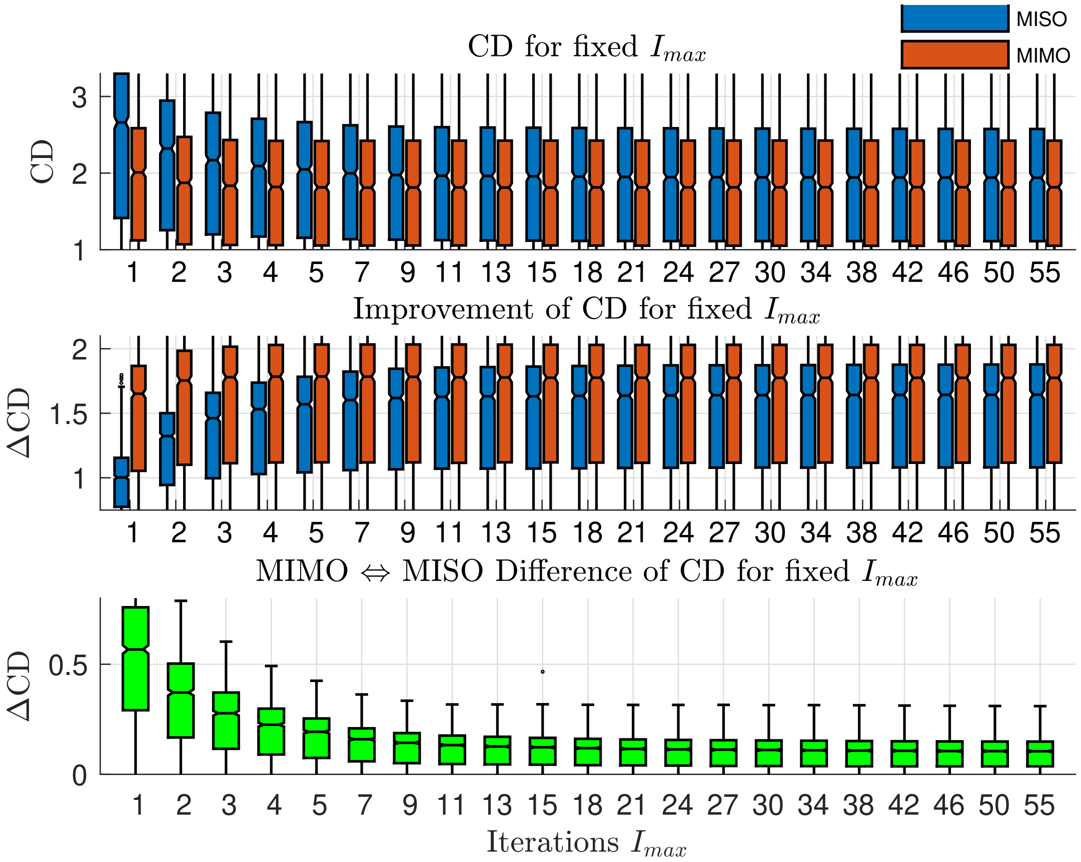

Figure 4.5: This boxplot shows the distributions of the CD of the beamformed signals and its improvements over the iterations I_{max}. The upper plot shows a comparison of the absolute CD between the MIMO-WPD and the MISO-WPD algorithm, the middle plot shows a comparison of the CD improvement compared to the noisy input signal between the MIMO-WPD and the MISO-WPD algorithm, whereas the lowest plot shows the difference of the two. In the lowest plot positive values indicate a larger CD improvement for MIMO-WPD.

4.3 Influence of Shape Parameter p

The shape parameter p determines the sparsity of the CGG sparse prior or the ℓ_p-norm cost function respectively as seen in fig. 2.3. Here the influence of this parameter is investigated whereby the proposed MIMO-WPD algorithm and the conventional MISO-WPD algorithm are performed on the Reverb Challenge dataset (see section 4.1.1) with the following shape parameter values

$$p = \{0, 0.1, 0.2, 0.3, \ldots, 1.9, 2\} \tag{4.3}$$

Since the interest lies in a comparison of the two algorithms after convergence a maximal number of iterations $I_{max} = 15$ was chosen for the following experiments, which is motivated by the results shown in section 4.2.1. In the following plots

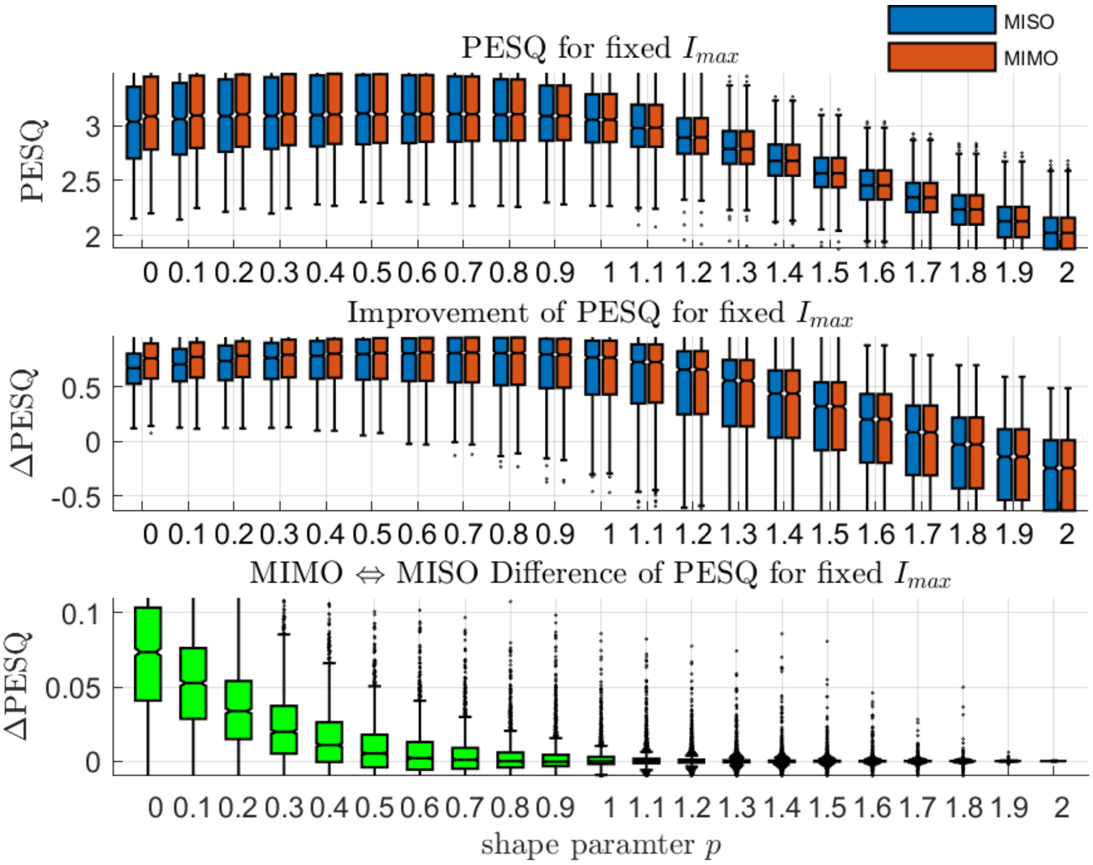

Figure 4.6: This boxplot shows the distributions of the PESQ of the beamformed signals and its improvements over the shape parameter p. The upper plot shows a comparison of the absolute PESQ between the MIMO-WPD and the MISO-WPD algorithm, the middle plot shows a comparison of the PESQ improvement compared to the noisy input signal between the MIMO-WPD and the MISO-WPD algorithm, whereas the lowest plot shows the difference of the two. In the lowest plot positive values indicate a larger PESQ improvement for MIMO-WPD.

(fig. 4.6, fig. 4.7, fig. 4.8) it can be observed, that a shape parameter of $p > 1$ leads to a performance reduction in all three objective speech quality measures. Comparing all of the results reveals that a shape parameter value of $p \approx 0.5$ leads to the highest performances for MIMO-WPD in all three measures, whereas MISO-WPD has its performance peak for PESQ at $p \approx 0.6$, for FWSSNR at $p \approx [0.4, 0.5]$ and for CD at $p \approx 0.4$. However it is very interesting that the largest medians of MIMO-WPD are only insignificantly larger than the largest medians of MISO-WPD, for all three objective measures. This is similar to the fact that for $p \approx [0.4, 0.6]$ the distributions of MIMO-WPD and MISO-WPD are not significantly different.

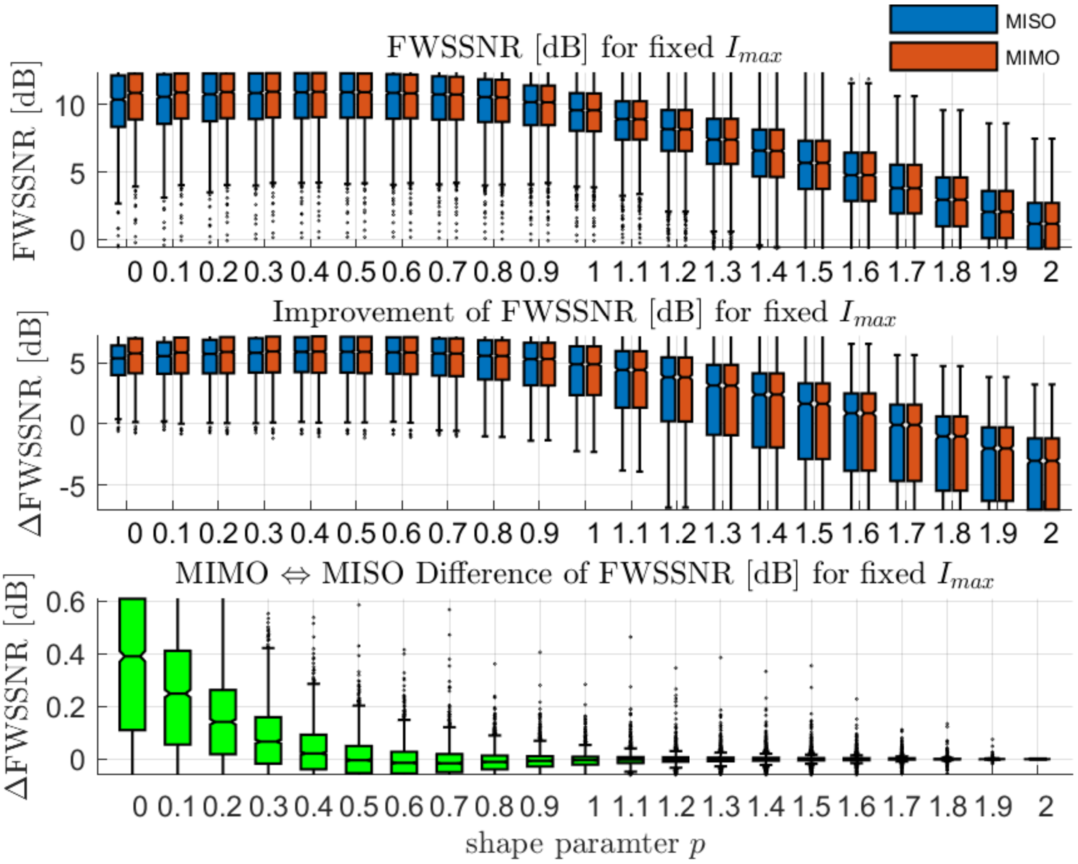

Figure 4.7: This boxplot shows the distributions of the FWSSNR of the beamformed signals and its improvements over the shape parameter p. The upper plot shows a comparison of the absolute FWSSNR between the MIMO-WPD and the MISO-WPD algorithm, the middle plot shows a comparison of the FWSSNR improvement compared to the noisy input signal between the MIMO-WPD and the MISO-WPD algorithm, whereas the lowest plot shows the difference of the two. In the lowest plot positive values indicate a larger FWSSNR improvement for MIMO-WPD.

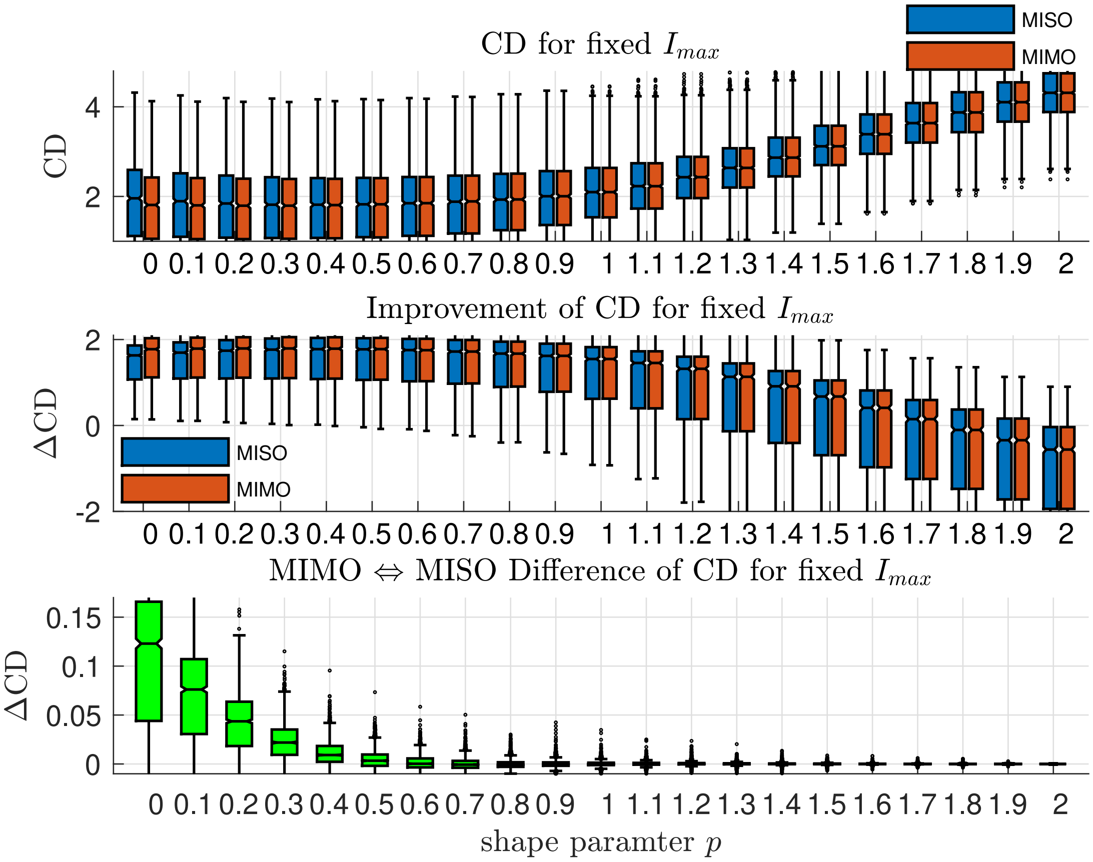

Figure 4.8: This boxplot shows the distributions of the CD of the beamformed signals and its improvements over the shape parameter p. The upper plot shows a comparison of the absolute CD between the MIMO-WPD and the MISO-WPD algorithm, the middle plot shows a comparison of the CD improvement compared to the noisy input signal between the MIMO-WPD and the MISO-WPD algorithm, whereas the lowest plot shows the difference of the two. In the lowest plot positive values indicate a larger CD improvement for MIMO-WPD.

Chapter 5

Conclusion

In many everyday scenarios reverberation and noise have clear detrimental effects on the speech quality as described in chapter 1. Therefore a need arises to develop and improve dereverberation and denoising algorithms. Over the past years many algorithms performing either denoising, dereverberation, or a combination of both were proposed, of which some are described in chapter 2. Recently the WPD algorithm was proposed by Nakatani et al. [1], which is described in section 3.2. This work proposes a reformulation of this WPD algorithm using the ℓ_p-norm as cost function in chapter 3. This introduces the shape parameter p, which controls the degree of sparsity of the cost function. Finally it changes the update of the weights (variances) in the conventional WPD as described in section 3.1. In the next step the MISO version of the WPD algorithm is extended to a MIMO version in section 3.2, whereby it is shown that this extension only requires a small modification of the weight update. In chapter 4 we investigated the performance of the MIMO and MISO version of the WPD algorithm as well as the influence of the shape parameter p. The evaluation was performed on the Reverb Challenge dataset described in section 4.1.1, which contains simulated multi-channel noisy reverberant speech signals. In section 4.2.1 it was shown that MIMO-WPD with a shape parameter of $p = 0$ significantly outperforms MISO-WPD in terms of PESQ, FWSSNR and CD. In addition MIMO-WPD outperforms MISO-WPD in terms of convergence speed, which indicates less computing cost. However the investigation of the influence of the shape parameter p in section 4.3 revealed that the performance of MIMO-WPD and MISO-WPD is similar for optimal values of p chosen for each algorithm. This indicates that the two modifications of the weight update, which are the shape parameter p and an additional RTF-vector term leading to MIMO-WPD, are non-complimentary.

5.1 Outlook

There are many parts of the WPD algorithm, which can be further investigated. One aspect is the mentioned correlation between the shape parameter modification and the RTF-vector term modification of the weight update. Other interesting directions include:

- Developing an adaptive version of the proposed MIMO-WPD with shape parameter p based on [19–21], which would be able to perform online processing in real time applications and to obtain estimates of signal statistics at each time frame. The latter is very important for non-stationary signals, whereby the RIR and the noise covariance matrix are time-varying.
- Performing additional source separation with the MIMO-WPD algorithm. One approach could be merging this work with [41].
- Modification of the wMPDR beamforming stage in factorized WPD with a multi-frame approach, which could be done similar as in [23, 42].
- Investigation of the influences of other parameters, e.g. the prediction delay, prediction filter length or the choice of the transform with its frame length, frame shift and window function.

Bibliography

[1] Tomohiro Nakatani and Keisuke Kinoshita. “A unified convolutional beamformer for simultaneous denoising and dereverberation”. In: *IEEE Signal Processing Letters* 26.6 (2019), pp. 903–907.

[2] Matthias R Mehl et al. “Are women really more talkative than men?” In: *Science* 317.5834 (2007), pp. 82–82.

[3] Aniansson Gunnar. “Methods for assessing high frequency hearing loss in every-day listening situations”. In: *Acta Oto-Laryngologica* 77.sup320 (1974), pp. 1–50.

[4] AJ Duquesnoy and R Plomp. “Effect of reverberation and noise on the intelligibility of sentences in cases of presbyacusis”. In: *The Journal of the Acoustical Society of America* 68.2 (1980), pp. 537–544.

[5] Reinier Plomp. “Auditory handicap of hearing impairment and the limited benefit of hearing aids”. In: *The Journal of the Acoustical Society of America* 63.2 (1978), pp. 533–549.

[6] Createc Beat Kaufmann. *Reverberation - Room Impulse Response.* [Online; accessed September 10, 2020]. 2007. URL: `https://www.beat-kaufmann.com/images/reverb-parameter_a.jpg`.

[7] Heinrich Kuttruff. *Room acoustics.* Crc Press, 2016.

[8] John S Bradley, Hiroshi Sato, and M Picard. “On the importance of early reflections for speech in rooms”. In: *The Journal of the Acoustical Society of America* 113.6 (2003), pp. 3233–3244.

[9] Iris Arweiler and Jörg M Buchholz. “The influence of spectral characteristics of early reflections on speech intelligibility”. In: *The Journal of the Acoustical Society of America* 130.2 (2011), pp. 996–1005.

[10] Anna Warzybok et al. “Influence of early reflections on speech intelligibility under different noise conditions”. In: *Forum Acusticum 2011. Proceedings.* 2011, pp. 1149–1154.

[11] Anna Warzybok et al. “Effects of spatial and temporal integration of a single early reflection on speech intelligibility”. In: *The Journal of the Acoustical Society of America* 133.1 (2013), pp. 269–282.

[12] Tomohiro Nakatani et al. “Importance of energy and spectral features in Gaussian source model for speech dereverberation”. In: *2007 IEEE Workshop on Applications of Signal Processing to Audio and Acoustics*. IEEE. 2007, pp. 299–302.

[13] Marc Delcroix, Takafumi Hikichi, and Masato Miyoshi. “Precise dereverberation using multichannel linear prediction”. In: *IEEE Transactions on Audio, Speech, and Language Processing* 15.2 (2007), pp. 430–440.

[14] Tomohiro Nakatani et al. “Blind speech dereverberation with multi-channel linear prediction based on short time Fourier transform representation”. In: *2008 IEEE International Conference on Acoustics, Speech and Signal Processing*. IEEE. 2008, pp. 85–88.

[15] Tomohiro Nakatani et al. “Speech dereverberation based on variance-normalized delayed linear prediction”. In: *IEEE Transactions on Audio, Speech, and Language Processing* 18.7 (2010), pp. 1717–1731.

[16] Takuya Yoshioka and Tomohiro Nakatani. “Generalization of multi-channel linear prediction methods for blind MIMO impulse response shortening”. In: *IEEE Transactions on Audio, Speech, and Language Processing* 20.10 (2012), pp. 2707–2720.

[17] Ante Jukić et al. “Multi-channel linear prediction-based speech dereverberation with sparse priors”. In: *IEEE/ACM Transactions on Audio, Speech, and Language Processing* 23.9 (2015), pp. 1509–1520.

[18] Ante Jukić et al. “Group sparsity for MIMO speech dereverberation”. In: *2015 IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA)*. IEEE. 2015, pp. 1–5.

[19] Ante Jukić, Toon van Waterschoot, and Simon Doclo. “Adaptive speech dereverberation using constrained sparse multichannel linear prediction”. In: *IEEE Signal Processing Letters* 24.1 (2016), pp. 101–105.

[20] Jahn Heymann et al. “Frame-online DNN-WPE dereverberation”. In: *2018 16th International Workshop on Acoustic Signal Enhancement (IWAENC)*. IEEE. 2018, pp. 466–470.

[21] Tomohiro Nakatani and Keisuke Kinoshita. “Simultaneous Denoising and Dereverberation for Low-Latency Applications Using Frame-by-Frame Online Unified Convolutional Beamformer.” In: *INTERSPEECH*. 2019, pp. 111–115.

[22] Barry D Van Veen and Kevin M Buckley. “Beamforming: A versatile approach to spatial filtering”. In: *IEEE assp magazine* 5.2 (1988), pp. 4–24.

[23] Jacob Benesty and Yiteng Huang. “A single-channel noise reduction MVDR filter”. In: *2011 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE. 2011, pp. 273–276.

[24] Shmulik Markovich, Sharon Gannot, and Israel Cohen. “Multichannel eigenspace beamforming in a reverberant noisy environment with multiple interfering speech signals”. In: *IEEE Transactions on Audio, Speech, and Language Processing* 17.6 (2009), pp. 1071–1086.

[25] Shmulik Markovich-Golan, Sharon Gannot, and Walter Kellermann. “Performance analysis of the covariance-whitening and the covariance-subtraction methods for estimating the relative transfer function”. In: *2018 26th European Signal Processing Conference (EUSIPCO)*. IEEE. 2018, pp. 2499–2503.

[26] acoustic camera. *Delay-And-Sum-Beamforming im Zeitbereich [1/1]*. [Online; accessed September 15, 2020]. 2017. URL: https://www.acoustic-camera.com/de/dienstleistungen/noise-and-vibration-blog/delay-and-sum-beamforming-im-zeitbereich.html.

[27] Marc Delcroix et al. “Strategies for distant speech recognitionin reverberant environments”. In: *EURASIP Journal on Advances in Signal Processing* 2015.1 (2015), p. 60.

[28] Wenxing Yang et al. “Dereverberation with differential microphone arrays and the weighted-prediction-error method”. In: *2018 16th International Workshop on Acoustic Signal Enhancement (IWAENC)*. IEEE. 2018, pp. 376–380.

[29] Tomohiro Nakatani and Keisuke Kinoshita. “Maximum likelihood convolutional beamformer for simultaneous denoising and dereverberation”. In: *2019 27th European Signal Processing Conference (EUSIPCO)*. IEEE. 2019, pp. 1–5.

[30] Christoph Boeddeker et al. “Jointly optimal dereverberation and beamforming”. In: *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE. 2020, pp. 216–220.

[31] Thomas Dietzen et al. “Joint multi-microphone speech dereverberation and noise reduction using integrated sidelobe cancellation and linear prediction”. In: *2018 16th International Workshop on Acoustic Signal Enhancement (IWAENC)*. IEEE. 2018, pp. 221–225.

[32] Thomas Dietzen et al. “Comparative analysis of generalized sidelobe cancellation and multi-channel linear prediction for speech dereverberation and noise reduction”. In: *IEEE/ACM Transactions on Audio, Speech, and Language Processing* 27.3 (2018), pp. 544–558.

[33] Thomas Dietzen et al. “Integrated sidelobe cancellation and linear prediction Kalman filter for joint multi-microphone speech dereverberation, interfering speech cancellation, and noise reduction”. In: *IEEE/ACM Transactions on Audio, Speech, and Language Processing* 28 (2020), pp. 740–754.

[34] Timo Gerkmann and Richard C Hendriks. “Unbiased MMSE-based noise power estimation with low complexity and low tracking delay”. In: *IEEE Transactions on Audio, Speech, and Language Processing* 20.4 (2011), pp. 1383–1393.

[35] Kaare Brandt Petersen and Michael Syskind Pedersen. “The matrix cookbook, nov 2012”. In: 3274 (2012), p. 14. URL: `http://www2.imm.dtu.dk/pubdb/p.php`.

[36] Are Hjorungnes and David Gesbert. “Complex-valued matrix differentiation: Techniques and key results”. In: *IEEE Transactions on Signal Processing* 55.6 (2007), pp. 2740–2746.

[37] Keisuke Kinoshita et al. “The REVERB challenge: A common evaluation framework for dereverberation and recognition of reverberant speech”. In: *2013 IEEE Workshop on Applications of Signal Processing to Audio and Acoustics*. IEEE. 2013, pp. 1–4.

[38] Tony Robinson et al. “WSJCAMO: a British English speech corpus for large vocabulary continuous speech recognition”. In: *1995 International Conference on Acoustics, Speech, and Signal Processing*. Vol. 1. IEEE. 1995, pp. 81–84.

[39] ITU-T Recommendation. "Perceptual evaluation of speech quality (PESQ): An objective method for end-to-end speech quality assessment of narrowband telephone networks and speech codecs". In: *Rec. ITU-T P. 862* (2001).

[40] Yi Hu and Philipos C Loizou. "Evaluation of objective quality measures for speech enhancement". In: *IEEE Transactions on audio, speech, and language processing* 16.1 (2007), pp. 229–238.

[41] Tomohiro Nakatani et al. "Jointly optimal denoising, dereverberation, and source separation". In: *arXiv preprint arXiv:2005.09843* (2020).

[42] Markus Kallinger and Alfred Mertins. "Multi-channel room impulse response shaping-a study". In: *2006 IEEE International Conference on Acoustics Speech and Signal Processing Proceedings*. Vol. 5. IEEE. 2006, pp. V–V.

A.1 Affidavit

I hereby affirm in lieu of oath that I wrote this work independently and did not use any other sources and aids than those indicated. I also affirm that I have followed the general principles of scientific work and publication as laid down in the guidelines of good scientific practice of the Carl von Ossietzky University Oldenburg.

Date, Signature

Eidesstattliche Erklärung

Hiermit versichere ich an Eides statt, dass ich diese Arbeit selbstständig verfasst und keine anderen als die angegebenen Quellen und Hilfsmittel benutzt habe. Außerdem versichere ich, dass ich die allgemeinen Prinzipien wissenschaftlicher Arbeit und Veröffentlichung, wie sie in den Leitlinien guter wissenschaftlicher Praxis der Carl von Ossietzky Universität Oldenburg festgelegt sind, befolgt habe.

Datum, Unterschrift