Part III.
Dereverberation

Tomohiro Nakatani
Speech recording in reverberant environments

Dereverberation is needed to enhance the quality of recorded speech by reducing reverberation included in it.
Effect of reverberation

Non-reverberant speech captured by a headset

Reverberant speech captured by a distant mic

Speech becomes less intelligible and ASR becomes very hard

Haeb-Umbach and Nakatani, Speech Enhancement - Dereverberation
Table of contents in part III

- Goal of dereverberation
- Approaches to dereverberation
  - Signal processing based approaches
  - A DNN-based approach
- Integration of signal processing and DNN approaches
  - DNN-WPE
Goal of dereverberation: time domain

$$x[t] = \sum_{\tau=0}^{L-1} a[\tau] s[t - \tau] = \sum_{\tau=0}^{D-1} a[\tau] s[t - \tau] + \sum_{\tau=D}^{L-1} a[\tau] s[t - \tau]$$

Preserve

Direct sound + Early reflections

Desired signal $$d[t]$$

Reduce

Late reverberation

$$r[t]$$

Direct sound

Impulse response $$a[\tilde{\tau}]$$

$$\tilde{D} (=30-50 \text{ ms})$$

Early reflections

Late reverberation

[Bradley et al., 2003]
Model of reverberation: STFT domain

- Time domain convolution is approximated by frequency domain convolution at each frequency [Nakatani et al. 2008]
  - If frame shift \(<<\) analysis window (e.g., frame shift \(\leq\) analysis window/4)

\[
\begin{align*}
\text{STFT domain (1-ch)} & \quad x_{t,f} = \sum_{\tau=0}^{L-1} a_{\tau,f} s_{t-\tau,f} = \\
\text{STFT domain (multi-ch)} & \quad x_{t,f} = \sum_{\tau=0}^{L-1} a_{\tau,f} s_{t-\tau,f} = \\
\text{Desired signal} & \quad \sum_{\tau=0}^{D-1} a_{\tau,f} s_{t-\tau,f} + \\
\text{Late reverberation} & \quad \sum_{\tau=D}^{L-1} a_{\tau,f} s_{t-\tau,f}
\end{align*}
\]

Convolutional transfer function:
\[
a_{\tau,f} = (a_{1,\tau,f}, a_{2,\tau,f}, \ldots, a_{M,\tau,f})^\top \quad \text{for} \quad \tau = 0, \ldots, L - 1
\]
Approaches to dereverberation

- **Beamforming (multi-ch)**
  - Enhance desired signal from speaker direction
  - Mostly the same as denoising

- **Blind inverse filtering (multi-ch)**
  - Cancel late reverberation
  - Multi-channel linear prediction
    - Weighted prediction error (WPE) method

- **DNN-based spectral enhancement (1ch)**
  - Estimate clean spectrogram
  - Mostly the same as denoising autoencoder
Approaches to dereverberation

• Beamforming (multi-ch)
  – Enhance desired signal from speaker direction
  – Mostly the same as denoising

• Blind inverse filtering (multi-ch)
  – Cancel late reverberation
  – Multi-channel linear prediction
    • Weighted prediction error (WPE) method

• DNN-based spectral enhancement (1ch)
  – Estimate clean spectrogram
  – Mostly the same as denoising autoencoder
Dereverberation based on beamforming

- Time domain model of desired signal
  \[ d(\tilde{t}) = \sum_{\tilde{\tau}=0}^{\tilde{D}} a(\tilde{\tau}) s(\tilde{t} - \tilde{\tau}) \]

- Assume \( \tilde{D} \ll \) STFT window, then
  \[ d_{t,f} = a_f s_{t,f} \]
  \[ x_{t,f} = a_f s_{t,f} + r_{t,f} \]

Beamforming is applicable to reduce \( r_{t,f} \)

- Techniques for estimating spatial covariances, \( \Psi_{dd,f} \) and \( \Psi_{rr,f} \)
  - Maximum-likelihood estimator [Schwartz et al., 2016]
  - Eigen-value decomposition based estimator [Heymann, 2017b, Kodrasi and Doclo, 2017, Nakatani et al., 2019a]
Approaches to dereverberation

- **Beamforming (multi-ch)**
  - Enhance desired signal from speaker direction
  - Mostly the same as denoising

- **Blind inverse filtering (multi-ch)**
  - Cancel late reverberation
  - Multi-channel linear prediction
    - Weighted prediction error (WPE) method

- **DNN-based spectral enhancement (1ch)**
  - Estimate clean spectrogram
  - Mostly the same as denoising autoencoder

Haeb-Umbach and Nakatani, Speech Enhancement - Dereverberation
What is inverse filtering

Clean speech $s_{f,t}$

RIRs

Reverberant speech (multi-ch) $x_{t,f}$

Inverse filter

Dereverberated speech (multi-ch) $s_{f,t}$ or $d_{f,t}$

Viewed as linear transformation (=matrix multiplication)

Inversion

Viewed as matrix inversion

Haeb-Umbach and Nakatani, Speech Enhancement - Dereverberation
Represent RIR convolution by matrix multiplication

1-ch representation

\[
\begin{pmatrix}
x_{m,t,f} \\
x_{m,t-1,f} \\
\vdots \\
x_{m,t-K,f}
\end{pmatrix}
= 
\begin{pmatrix}
a_{m,0,f} & a_{m,1,f} & \cdots & a_{m,L-1,f} & 0 & \cdots & 0 \\
0 & a_{m,0,f} & a_{m,1,f} & \cdots & a_{m,L-1,f} & \vdots & \vdots \\
\vdots & \vdots & \ddots & \ddots & \ddots & \vdots & \vdots \\
0 & \cdots & 0 & a_{m,0,f} & a_{m,1,f} & \cdots & a_{m,L-1,f}
\end{pmatrix}
\begin{pmatrix}
st_{t,f} \\
st_{t-1,f} \\
\vdots \\
st_{t-K_0,f}
\end{pmatrix}
\]

\[\bar{x}_{m,t,f} \in \mathbb{C}^K, \quad H_{m,f} \in \mathbb{C}^{K \times K_0}, \quad \bar{s}_{t,f} \in \mathbb{C}^{K_0}\]

\[\bar{x}_{m,t,f} = H_{m,f} \bar{s}_{m,t,f}\]

\[K_0 = L + K - 1\]

Multi-ch representation

\[
\begin{pmatrix}
\bar{x}_{1,t,f} \\
\vdots \\
\bar{x}_{M,t,f}
\end{pmatrix}
= 
\begin{pmatrix}
H_{1,f} \\
\vdots \\
H_{M,f}
\end{pmatrix}
\begin{pmatrix}
\bar{s}_{t,f}
\end{pmatrix}
\]

\[\bar{x}_{t,f} \in \mathbb{C}^{KM}, \quad H_f \in \mathbb{C}^{KM \times K_0}\]

\[\bar{x}_{t,f} = H_f \bar{s}_{t,f}\]
Existence of inverse filter [Miyoshi and Kaneda, 1988]

- Given $H_f$, the inverse filter $\bar{W}_f$ should satisfy

$$\bar{W}_f^H H_f = I$$

$I$ : identity matrix

- Solution exists and is obtained as:

$$\bar{W}_f^H = (H_f^H H_f)^{-1} H_f^H$$

- When $H_f$ is full column rank (roughly $\#\text{mics}>1$)

How can we estimate $\bar{W}_f$ without knowing $H_f$?
Approaches to blind inverse filtering

- Blind RIR estimation + robust inverse filtering
  - Blind RIR estimation is still an open issue
    - Eigen-decomposition [Gannot, 2010]
    - ML estimation approaches [Juang and Nakatani, 2007, Schmid et al., 2012]
  - Robust inverse filtering
    - Regularization [Hikichi et al., 2007]
    - Partial multichannel equalization [Kodrasi et al., 2013]

- Blind and direct estimation of inverse filter
  - Multichannel linear prediction (LP) based methods
    - Prediction Error (PE) method [Abed-Meraim et al., 1997]
    - Delayed Linear Prediction [Kinoshita et al., 2009]
    - Weighted Prediction Error (WPE) method [Nakatani et al., 2010]
    - Multi-input multi-output (MIMO) WPE method [Yoshioka and Nakatani, 2012]
  - Higher-order decorrelation approaches
    - Kurtosis maximization [Gillespie et al., 2001]
Multichannel LP [Abed-meraim et al, 1997]

\[
\text{Predict} \quad \sum_{\tau=1}^{L} W_{\tau,f}^H x_{t-\tau,f}
\]

\[\hat{d}_{t,f} \text{ : direct signal} \]
\[\hat{r}_{t,f} \text{ : reverberation} \]

Current signal
\[x_{t,f} = \hat{d}_{t,f} + \hat{r}_{t,f}\]

Past signal (multi-ch)

Dereverberation:
\[\hat{d}_{t,f} = x_{t,f} - \sum_{\tau=1}^{L} \hat{W}_{\tau,f}^H x_{t-\tau,f}\]

Subtract predictable components from observation
Definition of multichannel LP

- Multichannel autoregressive model

\[
x_{t,f} = \sum_{\tau=1}^{L} W_{\tau,f}^{H} x_{t-\tau,f} + \hat{d}_{t,f}
\]

\[W_{\tau,f} \in \mathbb{C}^{M \times M}_T: \text{prediction matrices.}\]

- Assuming \( \hat{d}_{t,f} \) stationary white noise, ML solution becomes

\[\{ \hat{W}_{\tau,f} \} = \arg\min_{\{ W_{\tau,f} \}} \sum_{t} \left\| x_{t,f} - \sum_{\tau=1}^{L} W_{\tau,f}^{H} x_{t-\tau,f} \right\|_2^2\]

- With estimated \( W_{\tau} \), \( \hat{d}_{t,f} \) is estimated (= inverse filtering) as

\[\hat{d}_{t,f} = x_{t,f} - \sum_{\tau=1}^{L} \hat{W}_{\tau,f}^{H} x_{t-\tau,f}\]
Problems in conventional LP

• Speech is not stationary white noise
  – LP assumes the target signal $d$ to be temporally uncorrelated
  – Speech signal exhibits short-term correlation (30-50 ms)
    LP distorts the short-time correlation of speech
  – LP assumes the target signal $d$ to be stationary
  – Speech is not stationary for long-time duration (200-1000 ms)
    LP destroys the time structure of speech

• Solutions:
  – Use of a prediction delay [Kinoshita et al., 2009]
  – Use of a better speech model [Nakatani et al, 2010]
Delayed LP (DLP) [Kinoshita et al., 2009]

\[ \text{Predict} \sum_{\tau = D}^{L} \mathbf{W}_{\tau, f}^{H} \mathbf{x}_{t - \tau, f} \]

Current signal \( \mathbf{x}_{t, f} = \mathbf{d}_{t, f} + \mathbf{r}_{t, f} \)

Past signal (multi-ch)

Delay \( D (=30-50 \text{ ms}) \)

Unpredictable

Predictable

Delayed LP can only predict \( \mathbf{r}_{t, f} \) from past signal

Only reduce \( \mathbf{r}_{t, f} \)
Introduction of better source model
[Nakatani et al., 2010, Yoshioka et al., 2011]

- Model of desired signal: time-varying Gaussian (local Gaussian)

\[ p(d_{t,f}; \theta) = N_c(d_{t,f}; 0, \sigma_{t,f}^2 I) \quad \theta = \{\sigma_{t,f}^2\} : \text{source PSD} \]

- ML estimation for time-varying Gaussian source

\[
\{\hat{W}_{\tau,f}, \hat{\sigma}_{t,f}^2\} = \arg\max_{\{W_{\tau,f}, \sigma_{t,f}^2\}} \prod_t \frac{1}{\pi\sigma_{t,f}^2} \exp\left(\frac{-\|x_{t,f} - \sum_{\tau=0}^{L} W_{\tau,f}^H x_{t-\tau,f}\|_2^2}{\sigma_{t,f}^2}\right)
\]

Minimization of weighted prediction error (WPE)

Blind inverse filtering can be achieved based only on a few seconds of observation
Processing flow of WPE

\[
\hat{d}_{t,f} = x_{t,f} - \sum_{\tau=D}^{L} \hat{W}_{\tau,f}^H x_{t-\tau,f}
\]

\[
\hat{\sigma}_{f,t}^2 = \frac{1}{M} \| x_{t,f} - \sum_{\tau=D}^{L} \hat{W}_{\tau,f}^H x_{t-\tau,f} \|_2^2
\]

\[
\{ \hat{W}_{\tau,f} \} = \arg\min_{\{W_{\tau,f}\}} \sum_{t} \| x_{t,f} - \sum_{\tau=D}^{L} \hat{W}_{\tau,f}^H x_{t-\tau,f} \|_2^2
\]
Why WPE achieves inverse filtering?

\[
\sum_t \frac{\|x_{t,f} - \sum_{\tau=D}^{L} W_{\tau,f}^H x_{t-\tau,f}\|_2^2}{\sigma_{t,f}^2} = \sum_t \frac{\|d_{t,f} + r_{t,f} - \sum_{\tau=D}^{L} W_{\tau,f}^H x_{t-\tau,f}\|_2^2}{\sigma_{t,f}^2} \\
\geq \sum_t \frac{\|d_{t,f}\|_2^2}{\sigma_{t,f}^2}
\]

Assumption
\(d_{t,f}\) is not correlated with \(r_{t,f}\) and with \(\sum_{\tau=D}^{L} W_{\tau,f}^H x_{t-\tau,f}\)

Minimized when
\(r_{t,f} = \sum_{\tau=D}^{L} W_{\tau,f}^H x_{t-\tau,f}\)

Reverb
Prediction

Existence of \(W_{\tau,f}\) is guaranteed when the inverse filter exists

Haeb-Umbach and Nakatani, Speech Enhancement - Dereverberation
Extensions

• Elaboration of probabilistic models
  – Sparse prior for speech PSD [Jukic et al., 2015]
  – Bayesian estimation with student-T speech prior [Chetupalli and Sreenivas, 2019]

• Frame-by-frame online estimation
  – Recursive least square [Yoshioka et al., 2009], [Caroselli et al., 2017]
  – Kalman filter for joint denoising and dereverberation [Togami and Kawaguchi, 2013], [Braun and Habets, 2018], [Dietzen et al., 2018]
Approaches to dereverberation

- Beamforming (multi-ch)
  - Enhance desired signal while reducing late reverberation
  - Mostly the same as denoising

- Blind inverse filtering (multi-ch)
  - Cancel late reverberation
  - (Multi-channel) Linear prediction
    - Weighted prediction error method

- DNN-based spectral enhancement (1ch)
  - Estimate clean spectrogram
  - Mostly the same as denoising autoencoder
Neural networks based dereverberation

- Train neural networks based on huge amount of parallel data

Many variations are proposed depending on tasks (masking/regression), cost functions, and network structures

[Weninger et al., 2014, Williamson and Wang, 2017]
REVERB Challenge task [Kinoshita et al., 2016]

- **Task**
  - Speech enhancement
  - ASR

- **Acoustic conditions**
  - Reverberation (Reverberation time 0.2 to 0.7 s.)
  - Stationary noise (SNR ~ 20dB)
### Comparison of three approaches

<table>
<thead>
<tr>
<th></th>
<th>Simu data</th>
<th>Real data</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>FWSSNR</td>
<td>CD</td>
</tr>
<tr>
<td><strong>Observed</strong></td>
<td>3.62 dB</td>
<td>3.97 dB</td>
</tr>
<tr>
<td><strong>MVDR</strong></td>
<td>6.59 dB</td>
<td>3.43 dB</td>
</tr>
<tr>
<td><strong>WPE</strong></td>
<td>4.79 dB</td>
<td>3.74 dB</td>
</tr>
<tr>
<td><strong>WPE+MVDR</strong></td>
<td>7.30 dB</td>
<td><strong>3.01 dB</strong></td>
</tr>
<tr>
<td><strong>DNN (soft mask estimation)</strong></td>
<td><strong>7.52 dB</strong></td>
<td>3.11 dB</td>
</tr>
</tbody>
</table>

**FWSSNR:** Frequency-weighted segmental SNR  
**CD:** Cepstral distortion  
**PESQ:** Perceptual evaluation of speech quality  
**WER:** Word error rate (obtained with Kaldi REVERB baseline)
Demonstration

1. Headset
2. Observed
3. MVDR
4. WPE
5. WPE+MVDR
6. DNN
## Pros and cons of three approaches

<table>
<thead>
<tr>
<th></th>
<th>Pros</th>
<th>Cons</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Beamforming</strong></td>
<td>• Low computational complexity</td>
<td>• Less effective dereverberation</td>
</tr>
<tr>
<td></td>
<td>• Capable of simultaneous denoising and dereverberation</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• High contribution to ASR</td>
<td></td>
</tr>
<tr>
<td><strong>WPE</strong></td>
<td>• <strong>Effective dereverberation</strong></td>
<td>• No denoising capability</td>
</tr>
<tr>
<td></td>
<td>• High contribution to ASR</td>
<td>• Computationally demanding</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• <strong>Iteration required for source PSD estimation</strong></td>
</tr>
<tr>
<td><strong>Neural networks</strong></td>
<td>• Effective dereverberation (source PSD estimation with no iterations)</td>
<td>• Sensitive to mismatched condition</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Low contribution to ASR</td>
</tr>
</tbody>
</table>
DNN-WPE [Kinoshita et al., 2017, Heyman et al., 2019]

1. No iterative estimation \(\rightarrow\) Effective for online processing
2. DNN can be optimized jointly with an ASR system

Advantages

\[
\hat{\sigma}_{f,t}^2 = \text{DNN}(x_{t,f})
\]

\[
\{\hat{W}_{\tau,f}\} = \text{argmin}_{\{W_{\tau,f}\}} \sum_{t} \frac{\|x_{t,f} - \sum_{\tau=D}^{L} W_{\tau,f}^H x_{t-\tau,f}\|_2^2}{\hat{\sigma}_{t,f}^2}
\]
Effectiveness of DNN-WPE [Heymann et al., 2019]

Training of DNN-WPE
- PSD-loss: MSE of PSD estimates
- ASR-loss: cross entropy of acoustic mode (AM) output

<table>
<thead>
<tr>
<th></th>
<th>REVERB (real)</th>
<th>WSJ+VoiceHome</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Offline</td>
<td>Online</td>
</tr>
<tr>
<td>Unprocessed</td>
<td>17.6</td>
<td>24.3</td>
</tr>
<tr>
<td>WPE</td>
<td>13.0</td>
<td>16.2</td>
</tr>
<tr>
<td>DNN-WPE (PSD loss)</td>
<td><strong>10.8</strong></td>
<td>14.6</td>
</tr>
<tr>
<td>DNN-WPE (ASR loss)</td>
<td>11.8</td>
<td><strong>13.4</strong></td>
</tr>
</tbody>
</table>

Denoising are not performed, and different ASR backend is used.
Frame-online framework for simultaneous denoising and dereverberation

- **WPD*¹**: a convolutional beamformer integrates WPE, beamformer, and DNN-based mask estimation

---

*¹: Weighted Power minimization

**Distortionless response**

**Convolutional beamformer**

---

Presentation at Interspeech 2019: 12:40-13:00, Mon, Sep. 16

[Nakatani et al, 2019b]
Software

• WPE
  – Matlab p-code for iterative offline, and block-online processing
    http://www.kecl.ntt.co.jp/icl/signal/wpe/
  – Python code w/ and w/o tensorflow for iterative offline, block-online, and frame-online processing
    https://pypi.org/project/nara-wpe/

• WPE, DNN-WPE
  – Python code with pytorch for offline and frame-online processing
    https://github.com/nttcslab-sp/dnn_wpe

  • Joint optimization of beamforming and dereverberation with end-to-end ASR enabled with espnet (https://github.com/espnet/espnet/espnet)
Table of contents

1. Introduction by Tomohiro
2. Noise reduction by Reinhold
3. Dereverberation by Tomohiro

Break (30 min)

4. **Source separation** by Reinhold
5. Meeting analysis by Tomohiro
6. Other topics by Reinhold
7. Summary by Reinhold & Tomohiro

QA