Reverberant Speech Processing for Human Communication and Automatic Speech Recognition

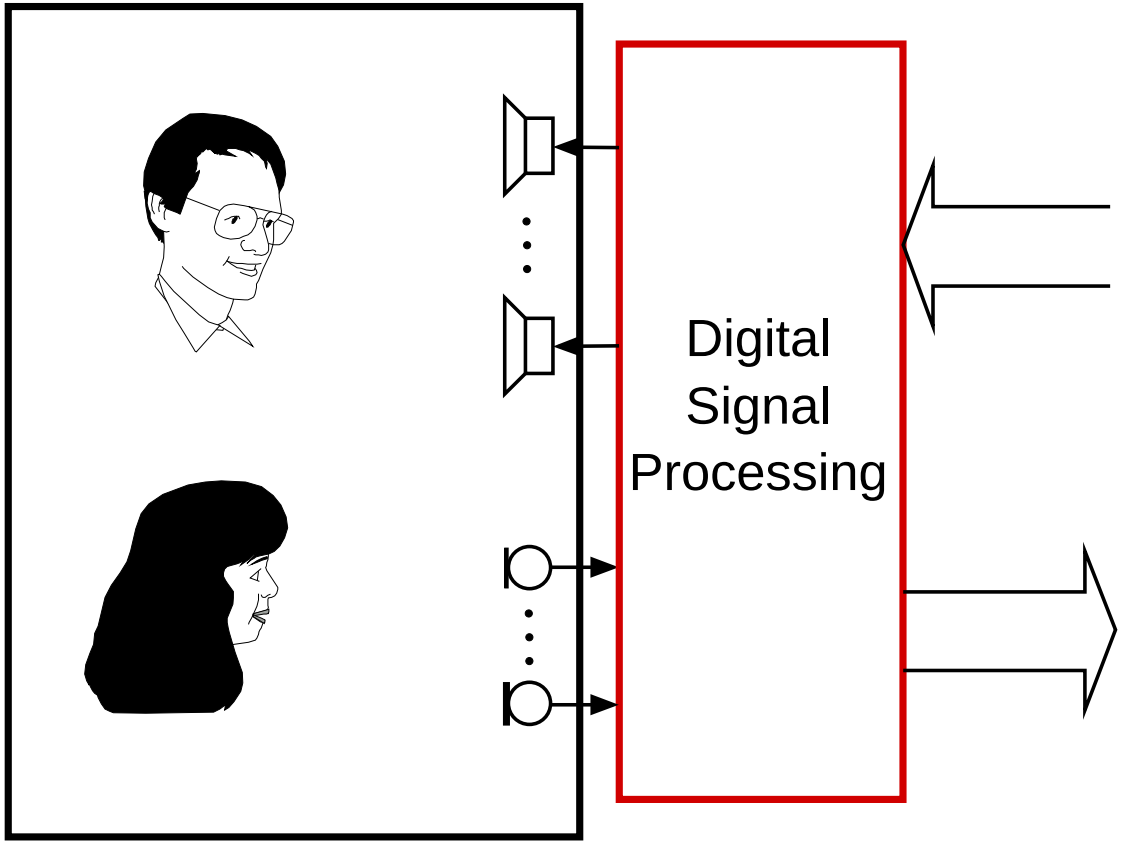
Tomohiro Nakatani, Armin Sehr, Walter Kellermann
nakatani.tomohiro@lab.ntt.co.jp, {sehr,wk}@LNT.de

NTT Communication Science Laboratories
LMS, University of Erlangen-Nuremberg

March 26, 2012

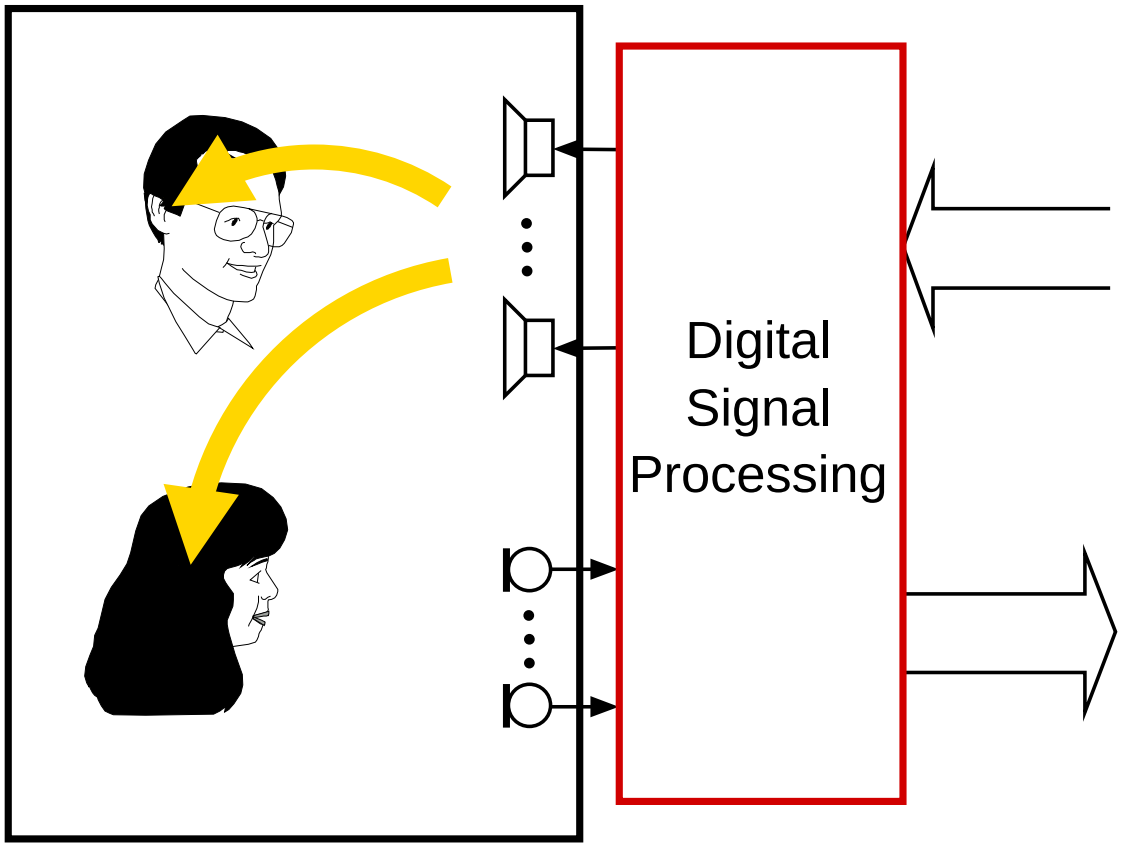
Generic Scenario:
Natural Interactive Human/Machine Interface

Mobile users, distant microphones/loudspeakers



Generic Scenario:
Natural Interactive Human/Machine Interface

Mobile users, distant microphones/loudspeakers

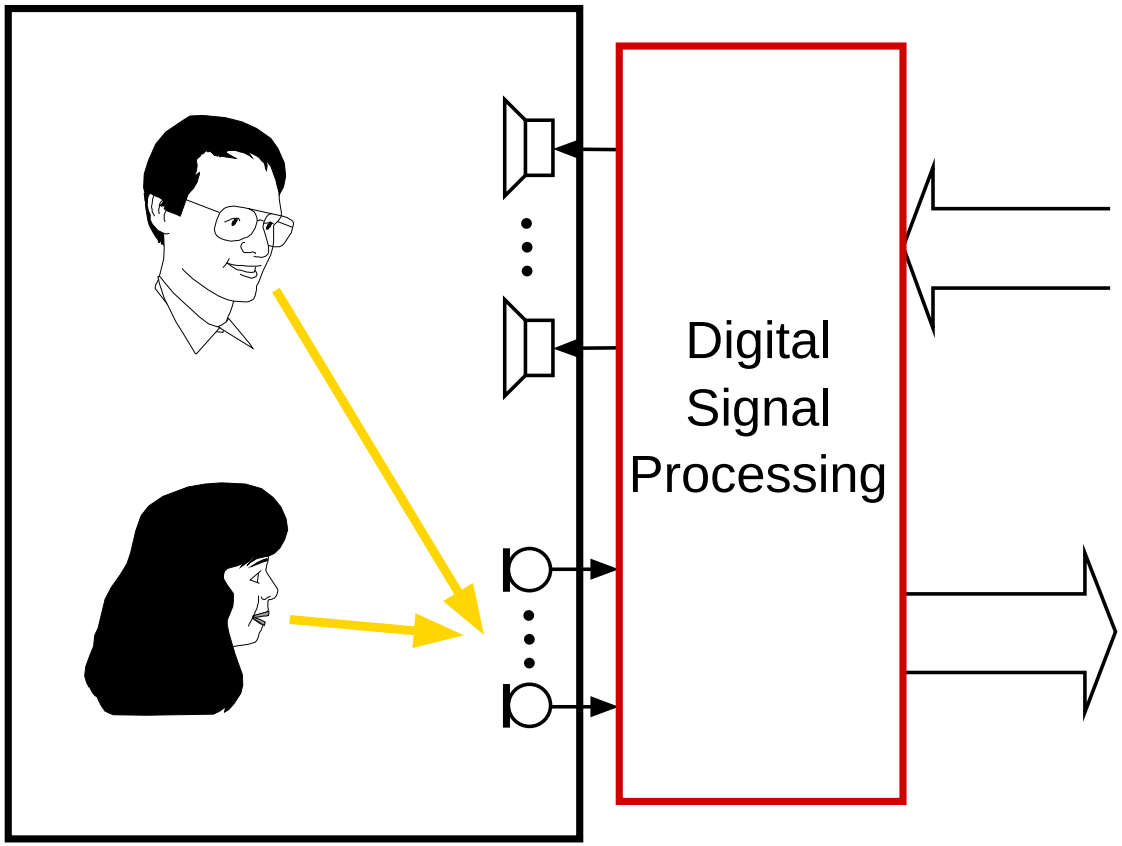


Tasks:

- Rendering - Reproduce desired signals at distant ears

Generic Scenario:
Natural Interactive Human/Machine Interface

Mobile users, distant microphones/loudspeakers

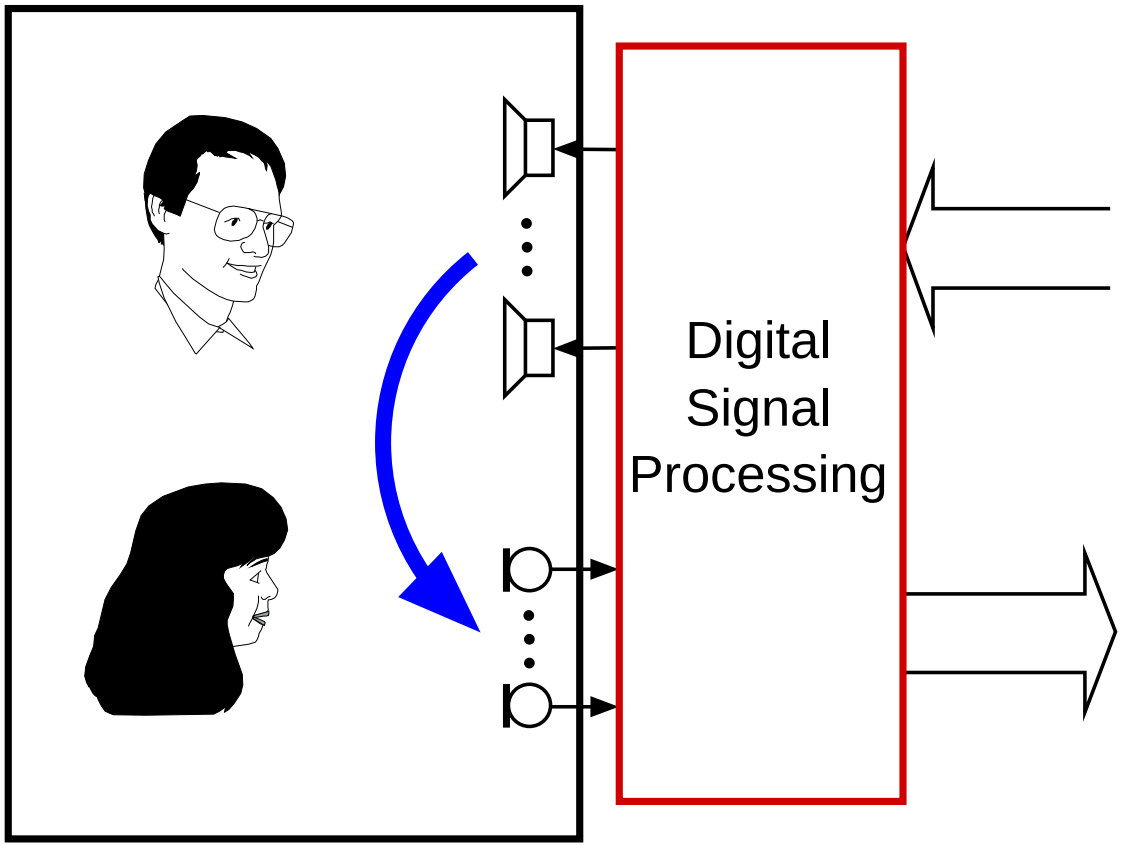


Tasks:

- Acquisition - Localize sources and capture clean signals from distance

Generic Scenario:
Natural Interactive Human/Machine Interface

Mobile users, distant microphones/loudspeakers



Tasks:

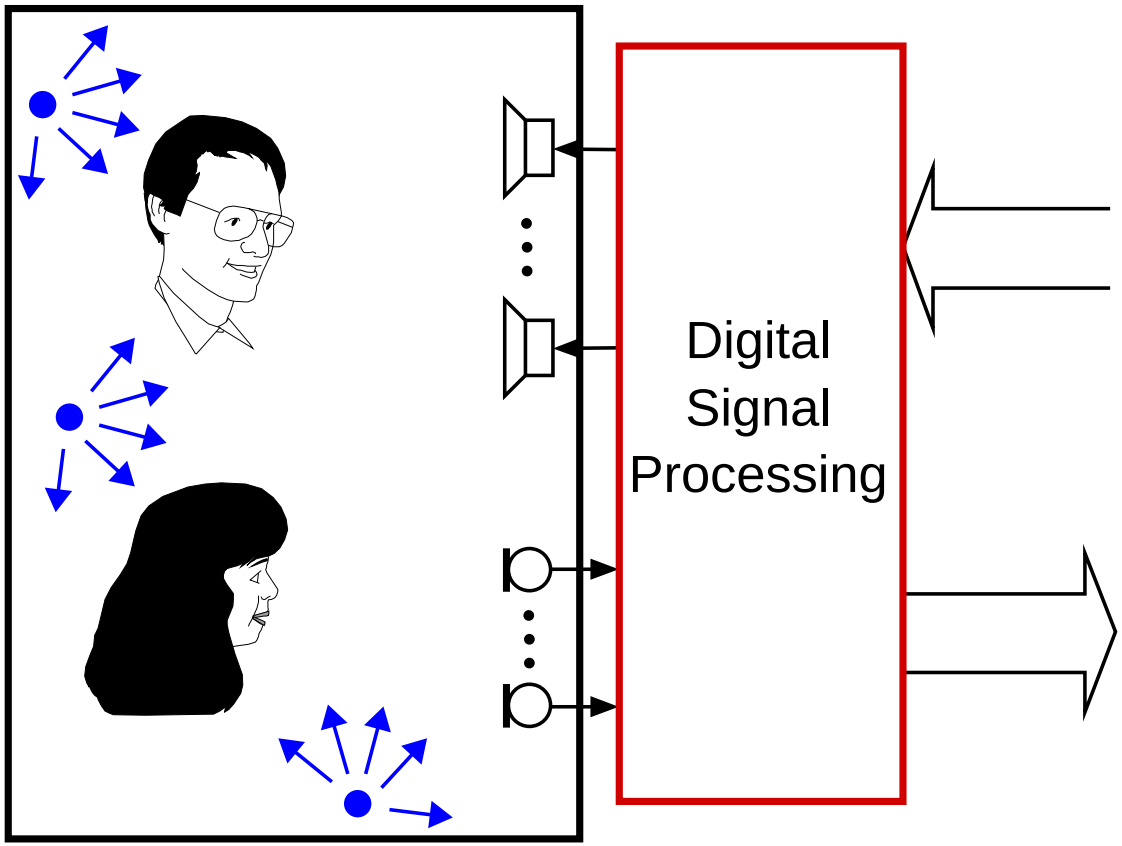
- Rendering - Reproduce desired signals at distant ears
- Acquisition - Localize sources and capture clean signals from distance

Challenges:

- Feedback of loudspeaker signals

Generic Scenario:
Natural Interactive Human/Machine Interface

Mobile users, distant microphones/loudspeakers



Tasks:

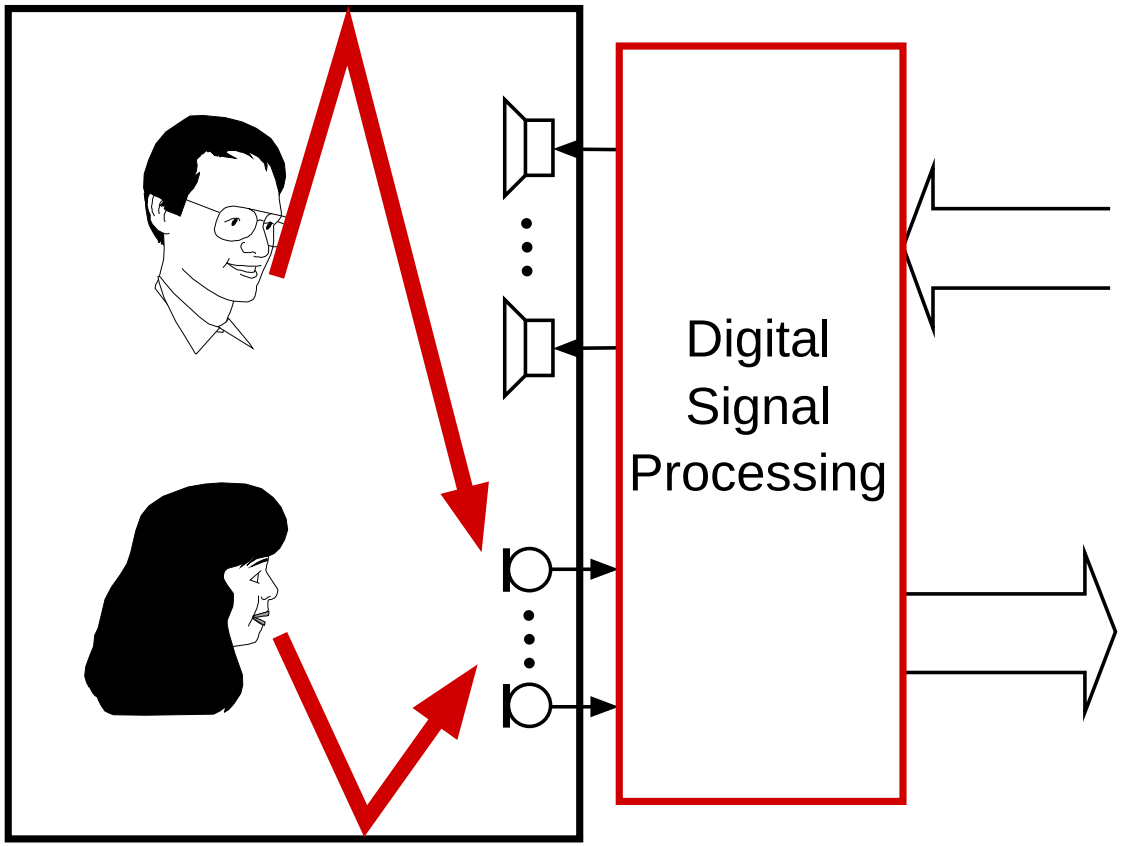
- Rendering - Reproduce desired signals at distant ears
- Acquisition - Localize sources and capture clean signals from distance

Challenges:

- Noise and interferers

Generic Scenario:
Natural Interactive Human/Machine Interface

Mobile users, distant microphones/loudspeakers



Tasks:

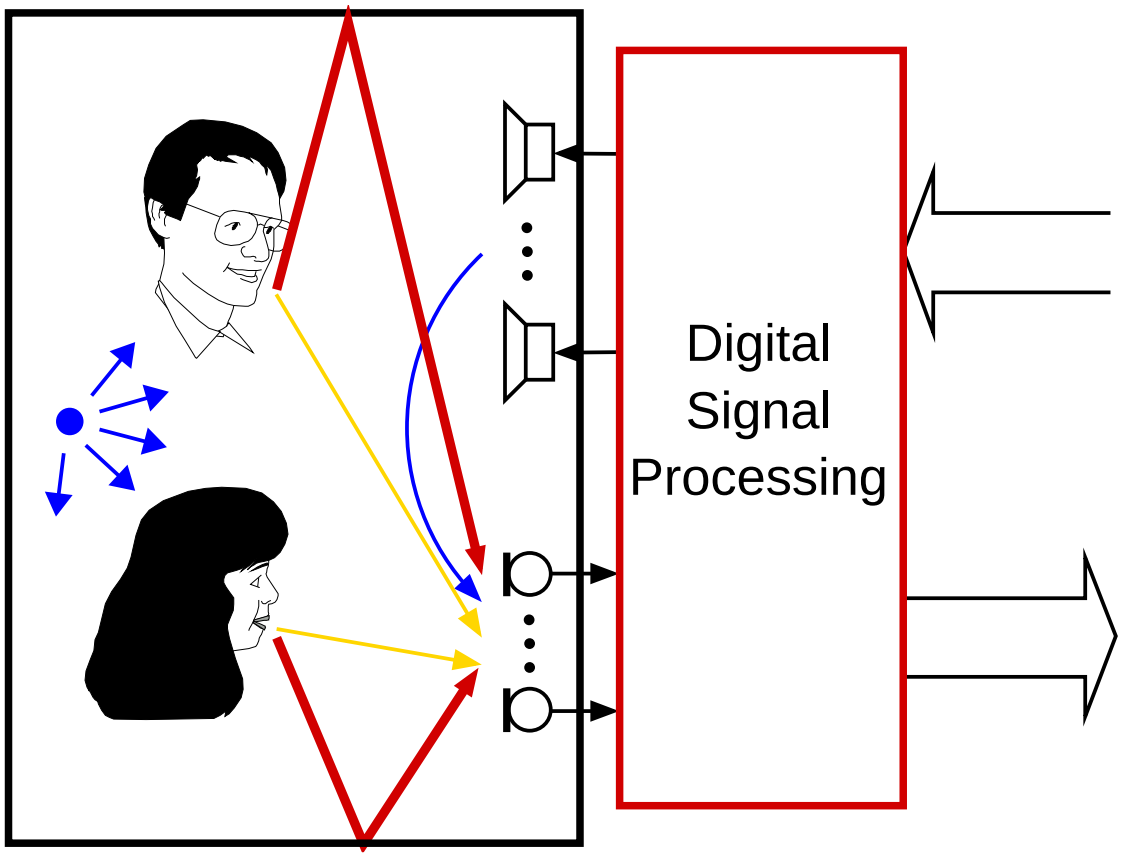
- Rendering - Reproduce desired signals at distant ears
- Acquisition - Localize sources and capture clean signals from distance

Challenges:

- **Reverberation**

Generic Scenario:
Natural Interactive Human/Machine Interface

Mobile users, distant microphones/loudspeakers



Tasks:

- Rendering - Reproduce desired signals at distant ears
- Acquisition - Localize sources and capture clean signals from distance

Challenges:

- Feedback of loudspeaker signals
- Noise and interferers
- **Reverberation**

Applications

- Hands-free equipment

  for telecommunication and natural human/machine interaction

  - for mobile phones / smart phones, mobile computing devices, PDAs
  - in car interiors (’command&control’, telecommunication, in-car communication, . . .)
  - for desktop computers, info-/edutainment terminals, interactive TV, game stations, simulators
  - for telepresence systems (offices,. . ., classrooms, . . ., auditoria)
  - for ambient communication (smart meeting rooms, smart homes, information kiosks, museums and exhibitions, . . .)
  - for voice-driven navigation systems in cars, operating rooms, . . .

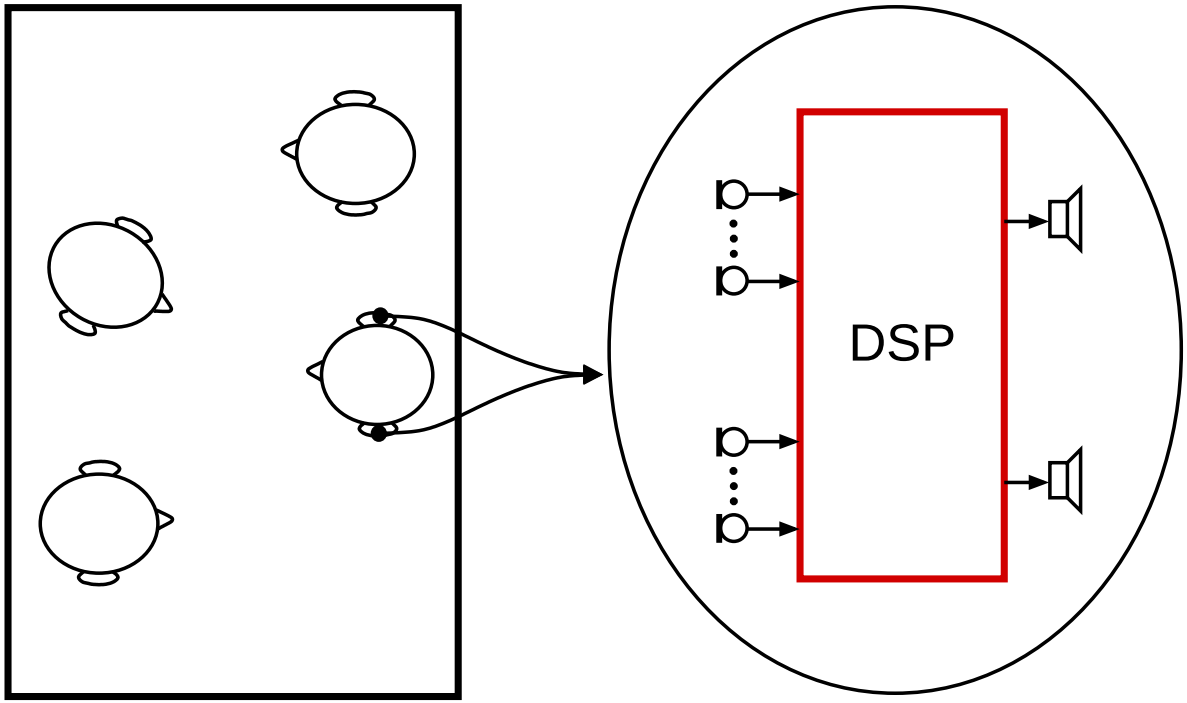
Applications (cont'd)

- Professional Audio
  - equipment for stages and recording studios
  - virtual acoustic environments (virtual concert halls, telepresence studios,. . .)

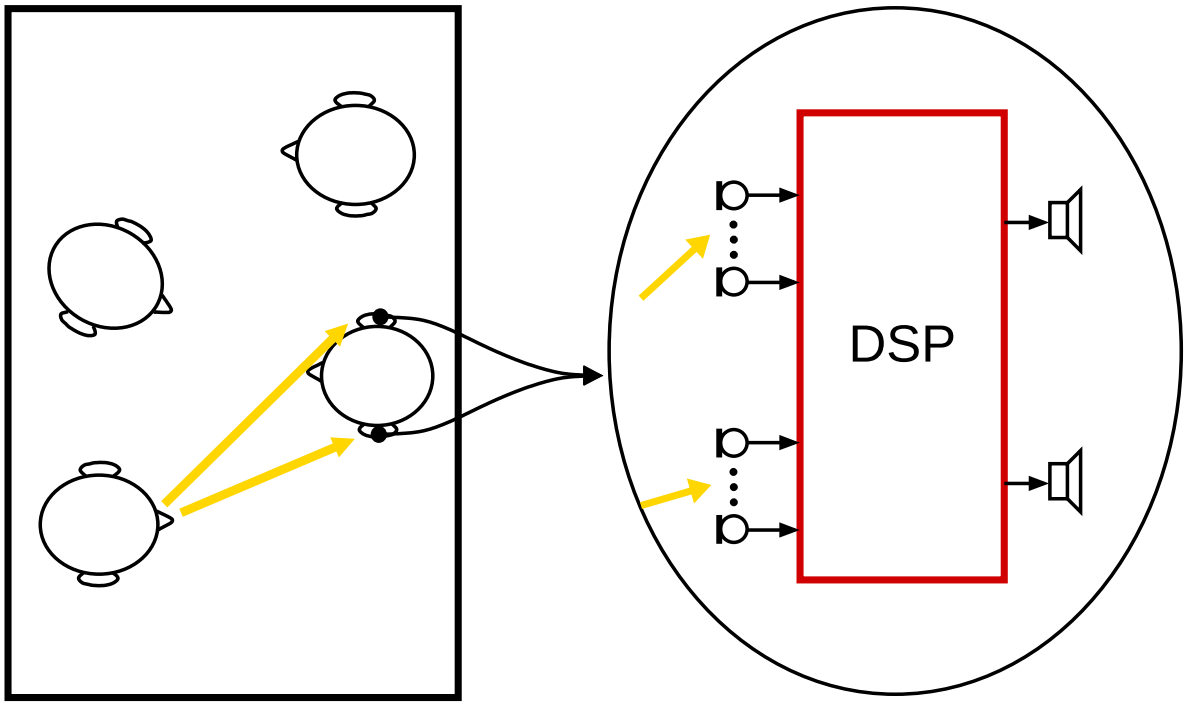
Applications (cont'd)

- Professional Audio
  - equipment for stages and recording studios
  - virtual acoustic environments (virtual concert halls, telepresence studios,. . .)
- Safety and Surveillance
  - acoustic displays in control centers, cockpits
  - monitoring in health care environments (advanced 'babyphones')
  - acoustic scene analysis (train stations, . . .)

Another Scenario: ’Listening devices’



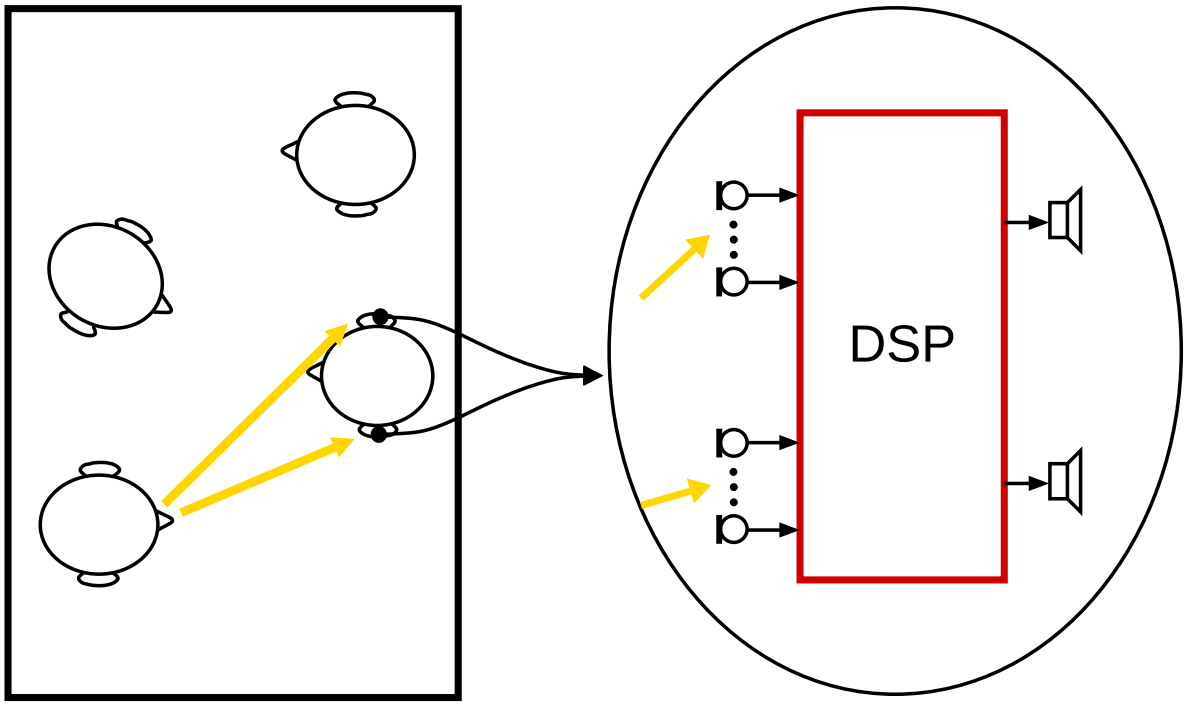
Another Scenario: ’Listening devices’



Tasks:

- Rendering - Reproduce undistorted signals with binaural cues

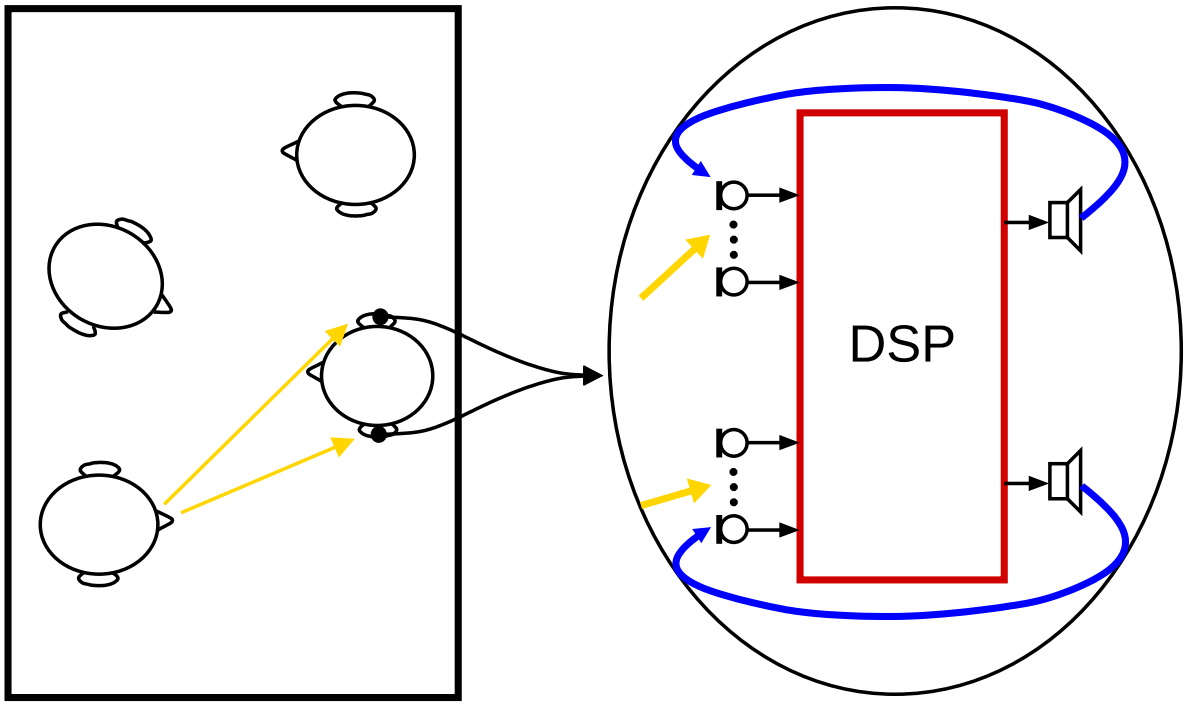
Another Scenario: ’Listening devices’



Tasks:

- Rendering - Reproduce undistorted signals with binaural cues
- Acquisition - Localize desired source(s) and enhance desired signal(s)

Another Scenario: ’Listening devices’



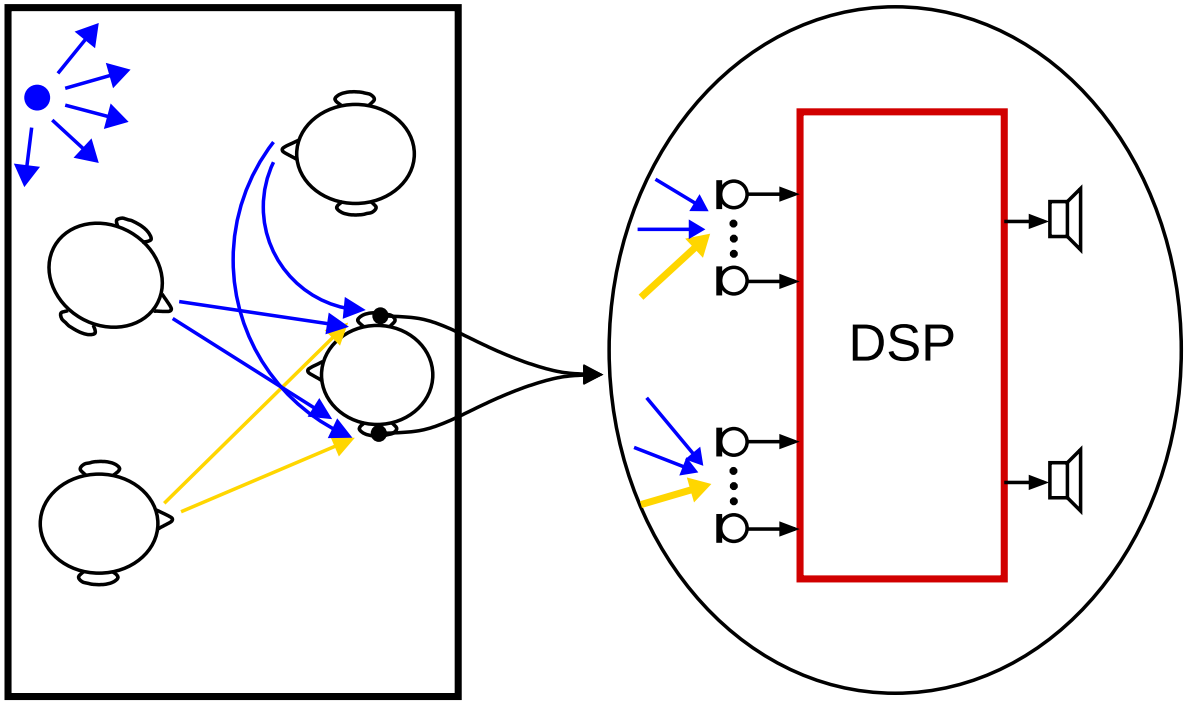
Tasks:

- Rendering - Reproduce undistorted signals with binaural cues
- Acquisition - Localize desired source(s) and enhance desired signal(s)

Challenges:

- Loudspeaker feedback (howling)

Another Scenario: ’Listening devices’



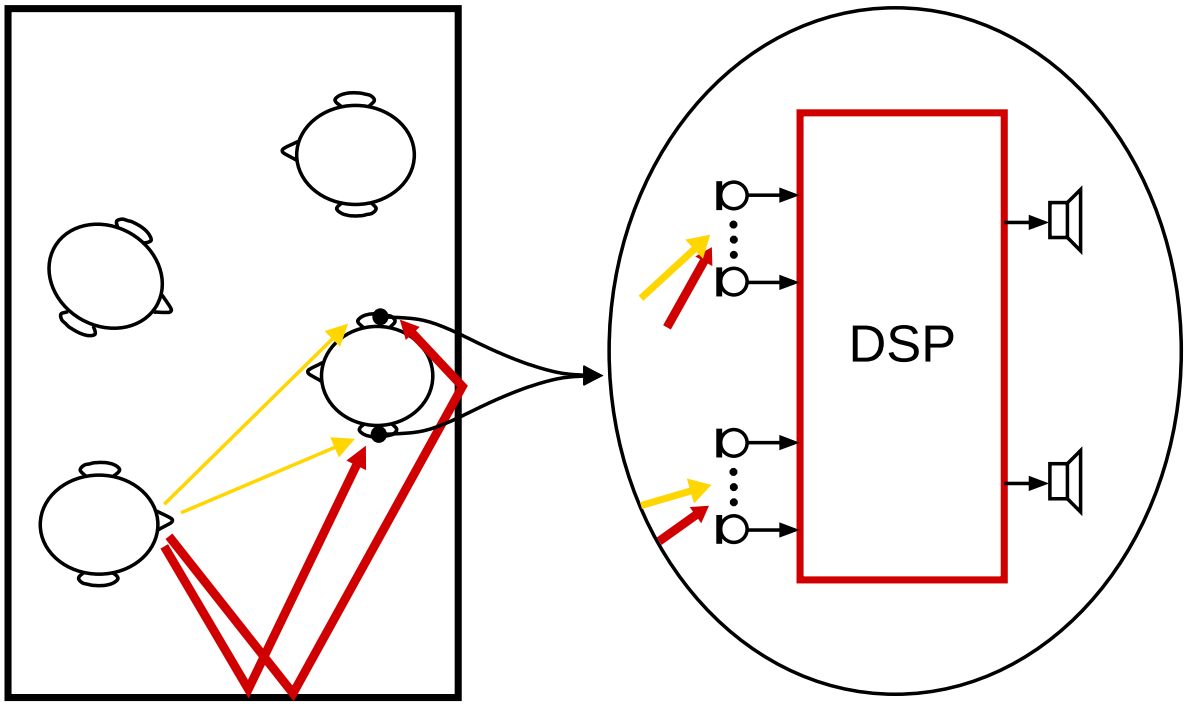
Tasks:

- Rendering - Reproduce undistorted signals with binaural cues
- Acquisition - Localize desired source(s) and enhance desired signal(s)

Challenges:

- Noise and interferers

Another Scenario: ’Listening devices’



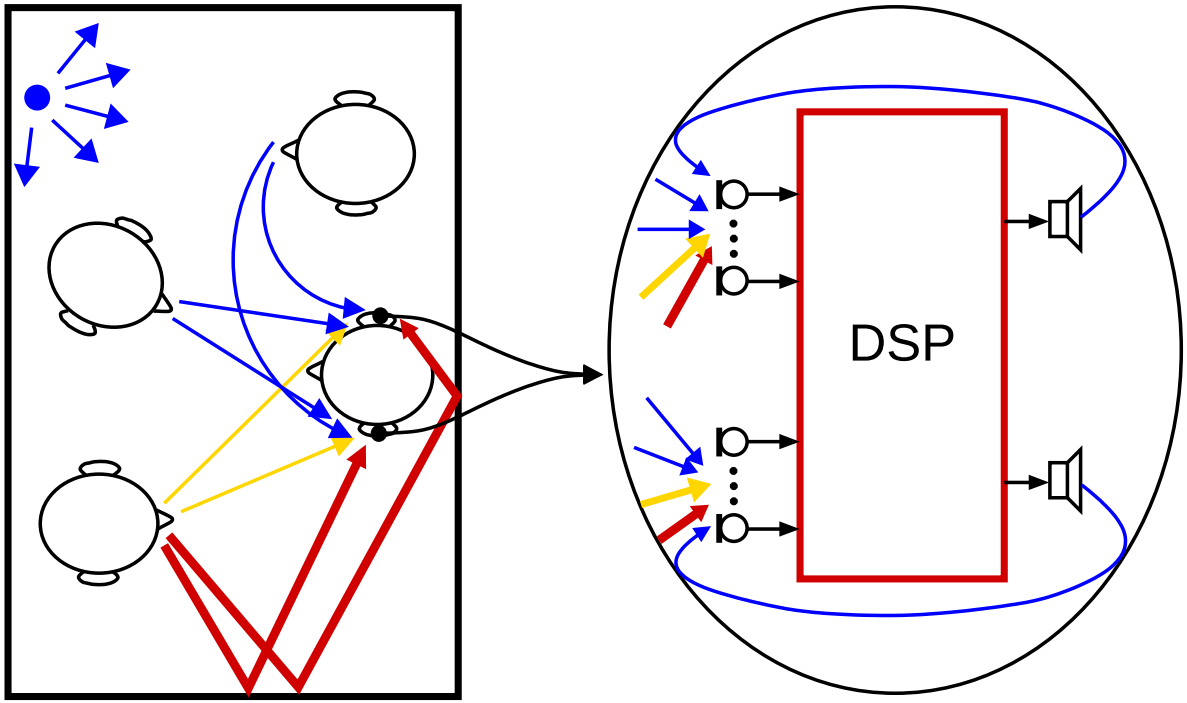
Tasks:

- Rendering - Reproduce undistorted signals with binaural cues
- Acquisition - Localize desired source(s) and enhance desired signal(s)

Challenges:

- **Reverberation**

Another Scenario: 'Listening devices'



Tasks:

- Rendering - Reproduce undistorted signals with binaural cues
- Acquisition - Localize desired source(s) and enhance desired signal(s)

Challenges:

- Loudspeaker feedback (howling)
- Noise and interferers
- **Reverberation**

Applications

- Hearing aids, of course
- Headsets, e.g., for
  - mobile phones, mobile computing devices, personal digital assistants
  - hearing protection in noisy environments (construction work, mining,. . .)
  - active noise cancellation systems
- . . .

Example 1: DICIT - an Interactive TV system

Voice-controlled home entertainment system (**EU project DICIT** 2005-2009; see e.g., Marquardt et al., 2009; **Youtube**)

Example 1: DICIT - an Interactive TV system

Voice-controlled home entertainment system (**EU project DICIT** 2005-2009; see e.g., Marquardt et al., 2009; **Youtube**)

featuring

- Multichannel AEC (GFDAF, Buchner/Benesty et al., 2003ff)
- Multibeamforming (Mabande et al., 2009; Kellermann, 1997)
- Source localization (GCF; Brutti et al., 2007)
- Speech/non-speech classification (Omologo, 2009)
- Noise-robust automatic speech recognition (ViaVoice, IBM 2009)

Example 1: DICIT - an Interactive TV system

Voice-controlled home entertainment system (**EU project DICIT** 2005-2009; see e.g., Marquardt et al., 2009; **Youtube**)

featuring

- Multichannel AEC (GFDAF, Buchner/Benesty et al., 2003ff)
- Multibeamforming (Mabande et al., 2009; Kellermann, 1997)
- Source localization (GCF; Brutti et al., 2007)
- Speech/non-speech classification (Omologo, 2009)
- Noise-robust automatic speech recognition (ViaVoice, IBM 2009)

Challenge: Reverberation for large source distances in more reverberant rooms

Example 2: Meeting recognition system

Recognize speech and other audio events

Who Spoke When, What, To-whom and How?

Real-time Meeting Browser

Example 3: Audio postproduction system

[Movies/TV creation]

Step1:
Sound&video recording (on location)

Step2:
Audio post-production
(de-noising, ***de-reverb***, sound effects)

Actor/actress

Microphone(s)

Overview

Part I: Introduction

Overview

Part I: Introduction

- Fundamentals
- Approaches

Overview

Part I: Introduction

Part II: Multichannel blind inverse filtering

- Example applications

Overview

Part I: Introduction

Part II: Multichannel blind inverse filtering

- Example applications
  - Professional audio post production
  - Meeting speech recognition with microphone arrays

Overview

Part I: Introduction

Part II: Multichannel blind inverse filtering

- Example applications
- Fundamentals: Dereverberation with inverse filtering

Overview

Part I: Introduction

Part II: Multichannel blind inverse filtering

- Example applications
- Fundamentals: Dereverberation with inverse filtering
  - What is ’inverse’ filtering?
  - Robust ’approximate’ inverse filtering

Overview

Part I: Introduction

Part II: Multichannel blind inverse filtering

- Example applications
- Fundamentals: Dereverberation with inverse filtering
- Blind inverse filtering

Overview

Part I: Introduction

Part II: Multichannel blind inverse filtering

- Example applications
- Fundamentals: Dereverberation with inverse filtering
- Blind inverse filtering
  - Overview of basic approaches
  - Closer look: multichannel linear prediction with time-varying source model

Overview

Part I: Introduction

Part II: Multichannel blind inverse filtering

- Example applications
- Fundamentals: Dereverberation with inverse filtering
- Blind inverse filtering
- Integration with blind source separation

Overview

Part I: Introduction

Part II: Multichannel blind inverse filtering

Part III: Robust ASR in reverberant environments

- Feature-based approaches

Overview

Part I: Introduction

Part II: Multichannel blind inverse filtering

Part III: Robust ASR in reverberant environments

- Feature-based approaches
  - Cepstral mean normalization
  - Model-based feature enhancement

Overview

Part I: Introduction

Part II: Multichannel blind inverse filtering

Part III: Robust ASR in reverberant environments

- Feature-based approaches
- Model-based approaches

Overview

Part I: Introduction

Part II: Multichannel blind inverse filtering

Part III: Robust ASR in reverberant environments

- Feature-based approaches
- Model-based approaches
  - Matched training
  - Multi-style training
  - Adaptive training
  - MAP and MLLR adaptation
  - Parametric adaptation tailored to reverberation
  - Frame-wise adaptation

Overview

Part I: Introduction

Part II: Multichannel blind inverse filtering

Part III: Robust ASR in reverberant environments

- Feature-based approaches
- Model-based approaches
- Decoder-based approaches

Overview

Part I: Introduction

Part II: Multichannel blind inverse filtering

Part III: Robust ASR in reverberant environments

- Feature-based approaches
- Model-based approaches
- Decoder-based approaches
  - Missing feature techniques
  - Uncertainty decoding

Overview

Part I: Introduction

Part II: Multichannel blind inverse filtering

Part III: Robust ASR in reverberant environments

- Feature-based approaches
- Model-based approaches
- Decoder-based approaches
- A generic approach: REMOS

Overview

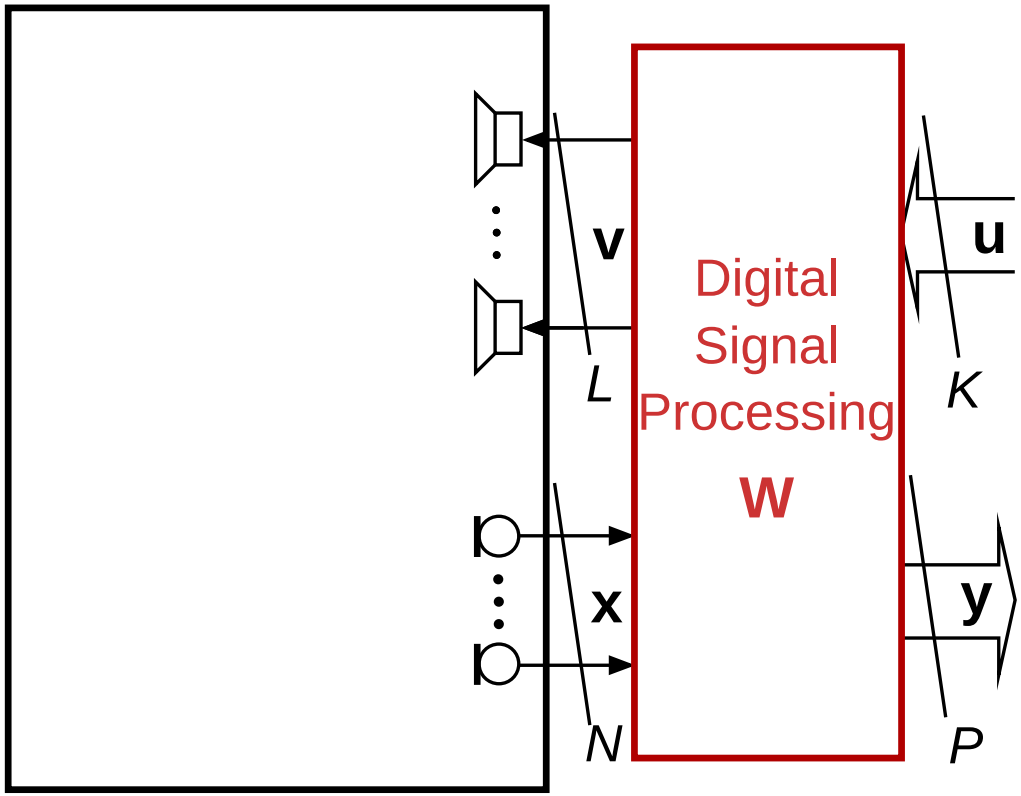
Part I: Introduction

Part II: Multichannel blind inverse filtering

Part III: Robust ASR in reverberant environments

Part IV: Summary, Conclusions, and Outlook

Fundamental Signal Processing Problems - Formulation

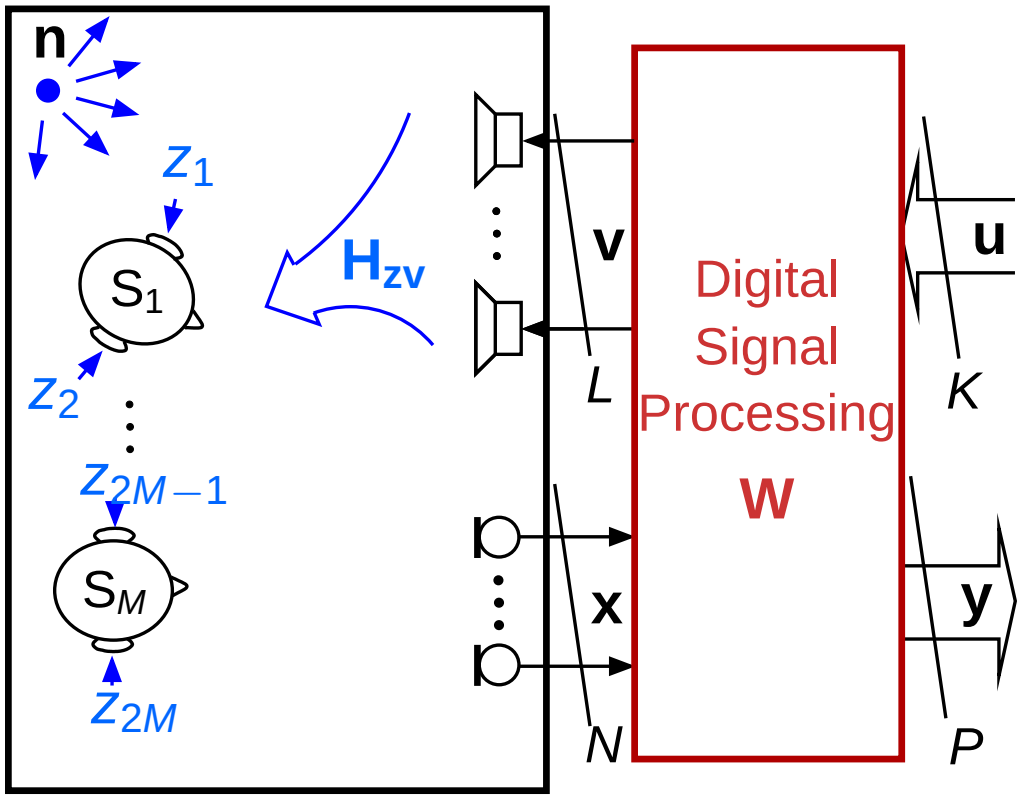


Linear MIMO system W ('multi-ple input/ multiple output'):

$$\begin{pmatrix} \mathbf{v} \\ \mathbf{y} \end{pmatrix} = \mathbf{W} * \begin{pmatrix} \mathbf{u} \\ \mathbf{x} \end{pmatrix}$$

$$= \begin{pmatrix} \mathbf{W}_{vu} & \mathbf{W}_{vx} \\ \mathbf{W}_{yu} & \mathbf{W}_{yx} \end{pmatrix} * \begin{pmatrix} \mathbf{u} \\ \mathbf{x} \end{pmatrix}$$

Fundamental Signal Processing Problems - Formulation



Linear MIMO system W ('multiple input/ multiple output'):

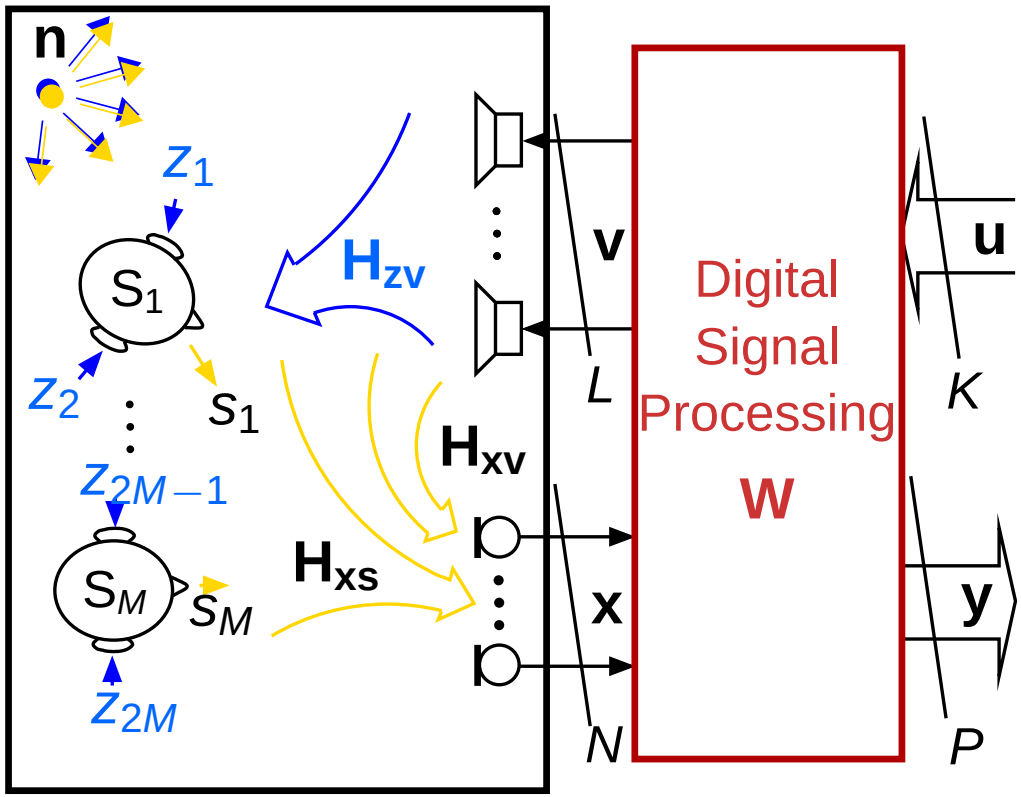
$$\begin{pmatrix} \mathbf{v} \\ \mathbf{y} \end{pmatrix} = \mathbf{W} * \begin{pmatrix} \mathbf{u} \\ \mathbf{x} \end{pmatrix}$$

$$= \begin{pmatrix} \mathbf{W_{vu}} & \mathbf{W_{vx}} \\ \mathbf{W_{yu}} & \mathbf{W_{yx}} \end{pmatrix} * \begin{pmatrix} \mathbf{u} \\ \mathbf{x} \end{pmatrix}$$

Listeners' signals:

$$\mathbf{z} = \mathbf{H_{zv}} * \mathbf{v} + \mathbf{n_z}$$

Fundamental Signal Processing Problems - Formulation



Linear MIMO system W ('multiple input/ multiple output'):

$$\begin{pmatrix} \mathbf{v} \\ \mathbf{y} \end{pmatrix} = \mathbf{W} * \begin{pmatrix} \mathbf{u} \\ \mathbf{x} \end{pmatrix} = \begin{pmatrix} \mathbf{W_{vu}} & \mathbf{W_{vx}} \\ \mathbf{W_{yu}} & \mathbf{W_{yx}} \end{pmatrix} * \begin{pmatrix} \mathbf{u} \\ \mathbf{x} \end{pmatrix}$$

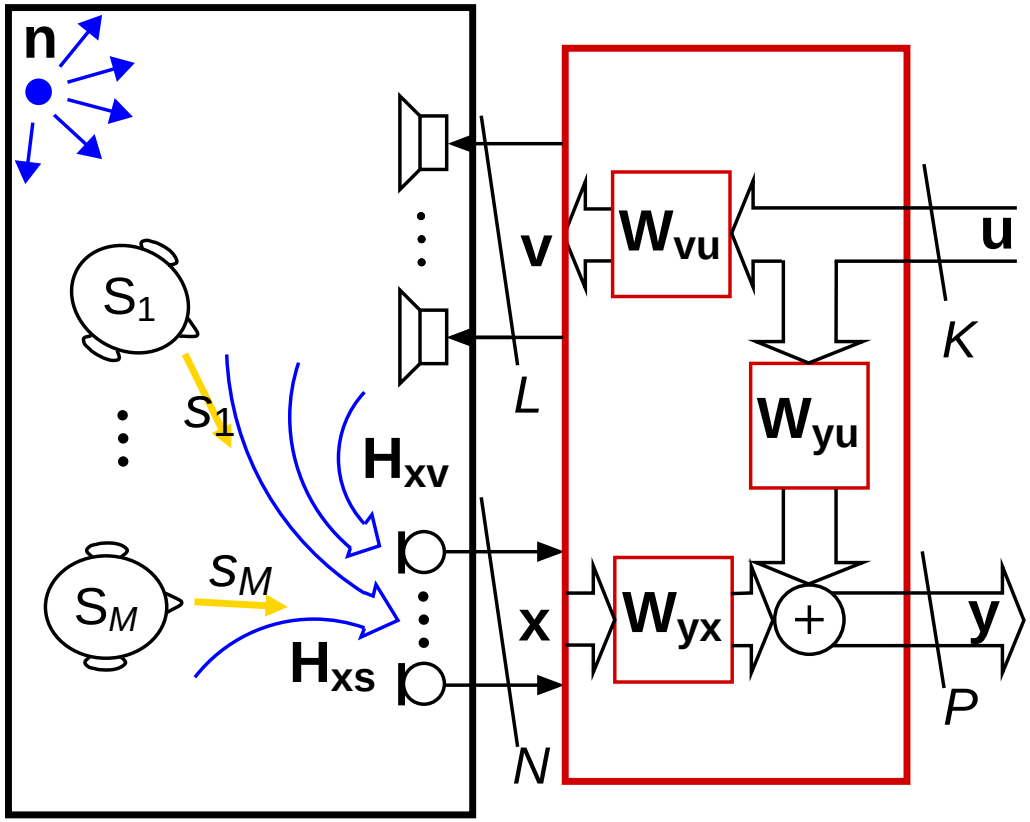
Listeners' signals:

$$\mathbf{z} = \mathbf{H_{zv}} * \mathbf{v} + \mathbf{n_z}$$

Microphone signals:

$$\mathbf{x} = \mathbf{H_{xs}} * \mathbf{s} + \mathbf{H_{xv}} * \mathbf{v} + \mathbf{n_x}$$

Fundamental Problems for Signal Acquisition

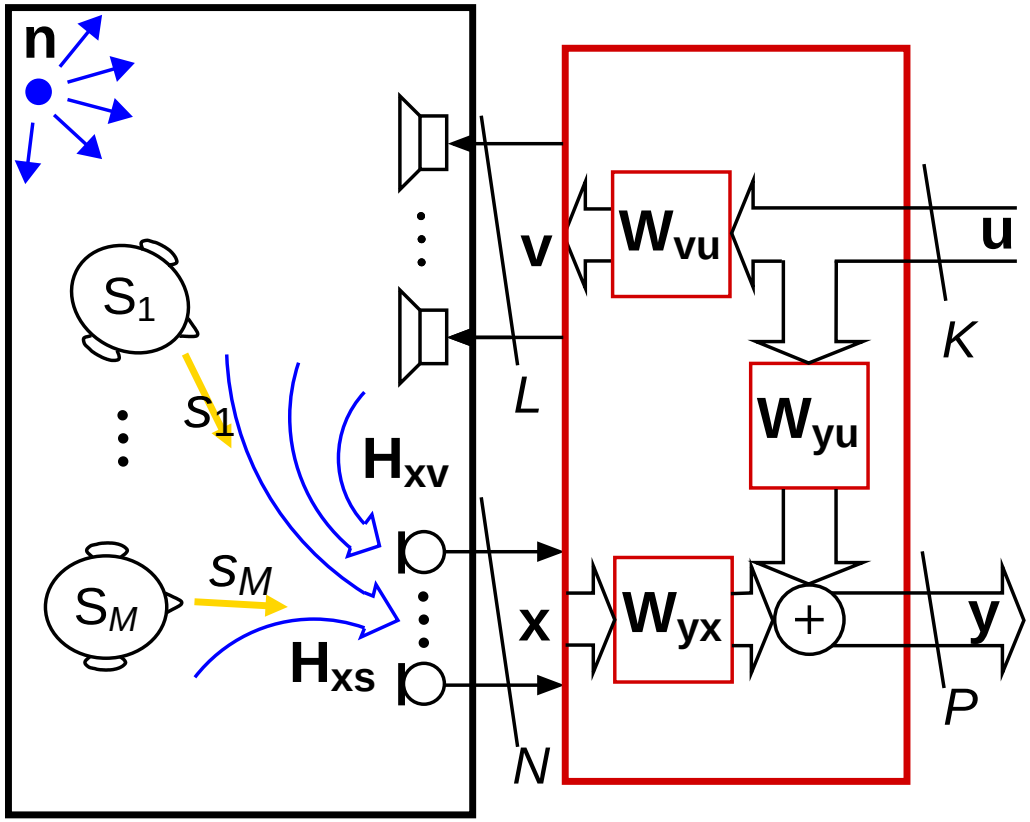


Goal: Undistorted source signals

$$\mathbf{y} = \mathbf{W_{yu}} * \mathbf{u} + \mathbf{W_{yx}} * \mathbf{x} \stackrel{!}{=} \mathbf{s} * \delta(k - k_0)$$

where $\mathbf{x} = \mathbf{H_{xs}} * \mathbf{s} + \mathbf{H_{xv}} * \mathbf{v} + \mathbf{n_x}$

Fundamental Problems for Signal Acquisition



Goal: Undistorted source signals

$$\mathbf{y} = \mathbf{W_{yu}} * \mathbf{u} + \mathbf{W_{yx}} * \mathbf{x} \stackrel{!}{=} \mathbf{s} * \delta(k - k_0)$$

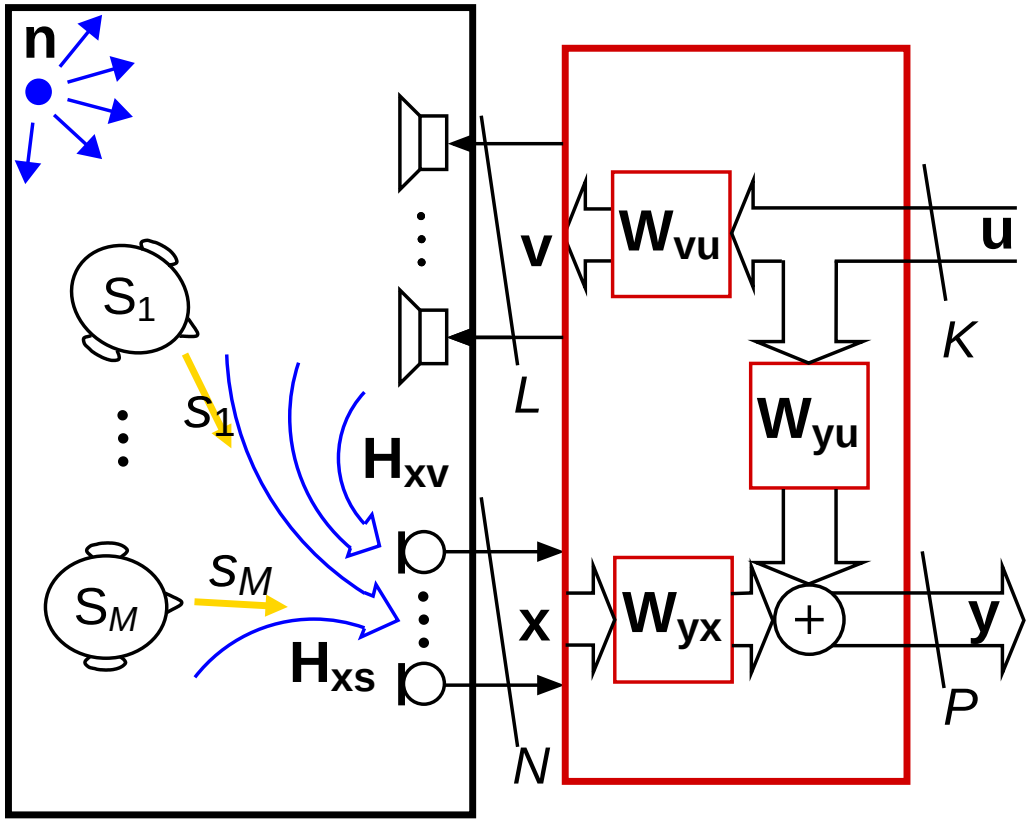
where $\mathbf{x} = \mathbf{H_{xs}} * \mathbf{s} + \mathbf{H_{xv}} * \mathbf{v} + \mathbf{n_x}$

3 Subproblems:

- Echo cancellation:

$$(\mathbf{W_{yu}} + \mathbf{W_{yx}} * \mathbf{H_{xv}} * \mathbf{W_{vu}}) * \mathbf{u} = \mathbf{0}$$

Fundamental Problems for Signal Acquisition



Goal: Undistorted source signals

$$\mathbf{y} = \mathbf{W_{yu}} * \mathbf{u} + \mathbf{W_{yx}} * \mathbf{x} \stackrel{!}{=} \mathbf{s} * \delta(k - k_0)$$

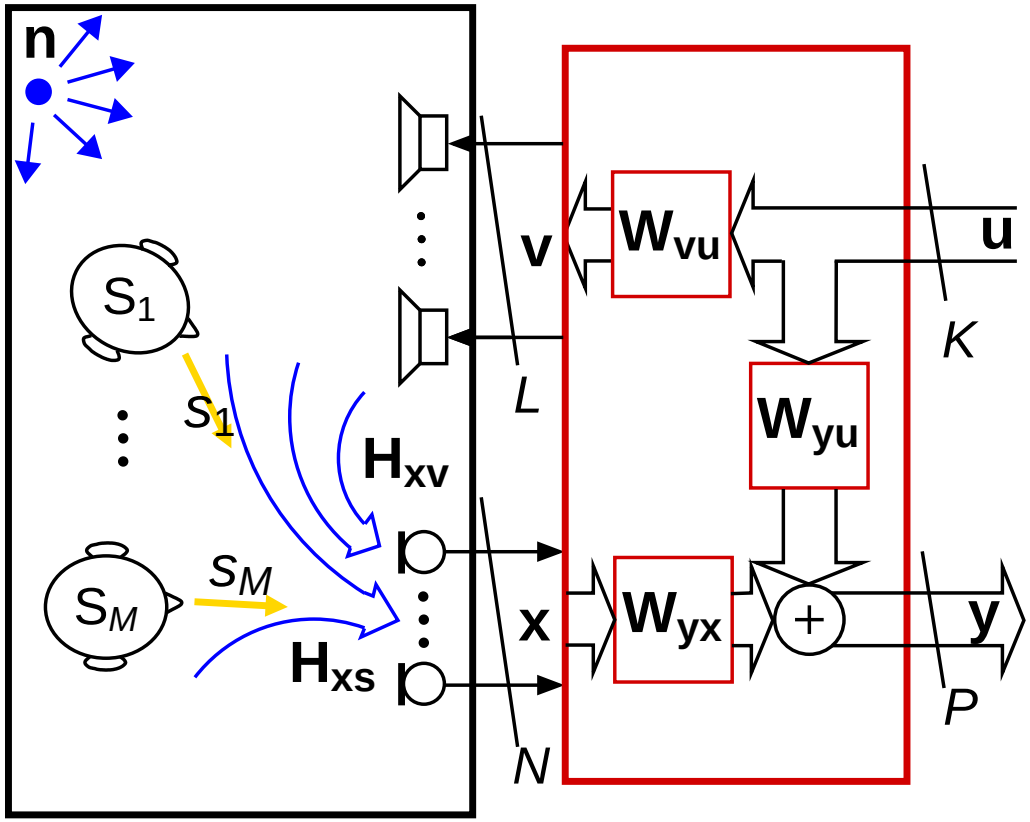
where $\mathbf{x} = \mathbf{H_{xs}} * \mathbf{s} + \mathbf{H_{xv}} * \mathbf{v} + \mathbf{n_x}$

3 Subproblems:

- Source separation and **dereverberation**:

$$\mathbf{W_{yx}} * \mathbf{H_{xs}} * \mathbf{s} = \mathbf{s} * \delta(k - k_0)$$

Fundamental Problems for Signal Acquisition



Goal: Undistorted source signals

$$\mathbf{y} = \mathbf{W_{yu}} * \mathbf{u} + \mathbf{W_{yx}} * \mathbf{x} \stackrel{!}{=} \mathbf{s} * \delta(k - k_0)$$

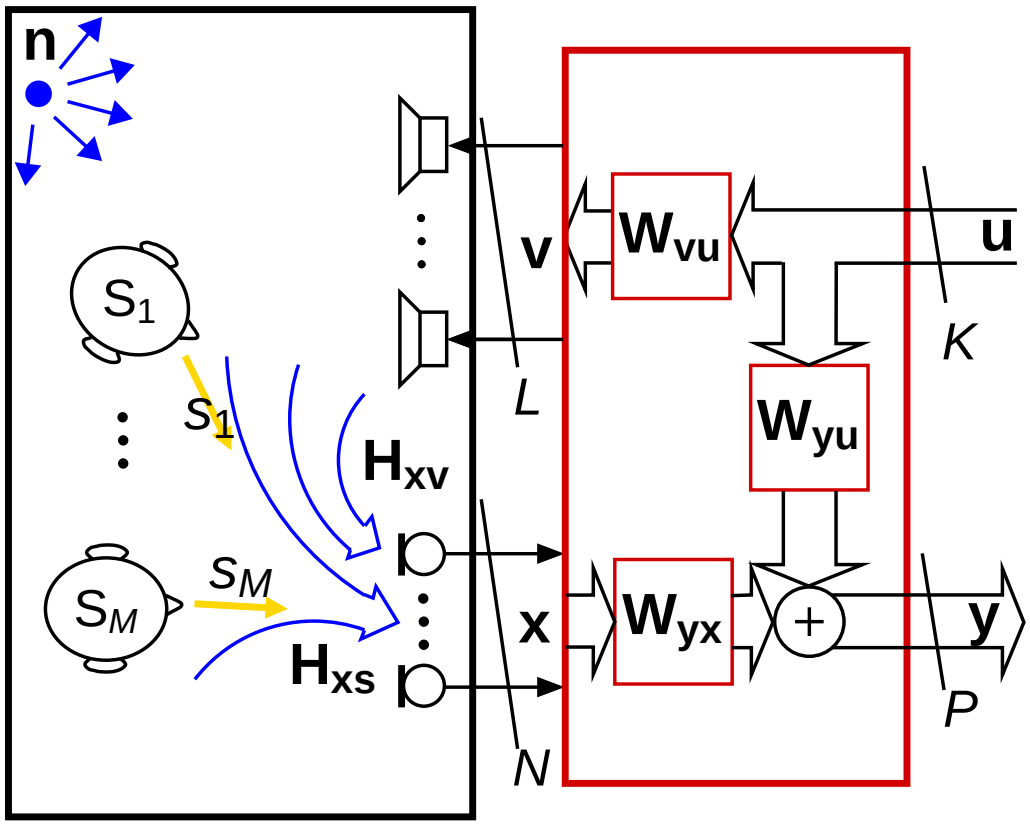
where $\mathbf{x} = \mathbf{H_{xs}} * \mathbf{s} + \mathbf{H_{xv}} * \mathbf{v} + \mathbf{n_x}$

3 Subproblems:

- Noise and interference suppression:

$$\mathbf{W_{yx}} * \mathbf{n_x} = \mathbf{0}$$

Fundamental Problems for Signal Acquisition



Goal: Undistorted source signals

$$\mathbf{y} = \mathbf{W_{yu}} * \mathbf{u} + \mathbf{W_{yx}} * \mathbf{x} \stackrel{!}{=} \mathbf{s} * \delta(k - k_0)$$

where $\mathbf{x} = \mathbf{H_{xs}} * \mathbf{s} + \mathbf{H_{xv}} * \mathbf{v} + \mathbf{n_x}$

3 Subproblems:

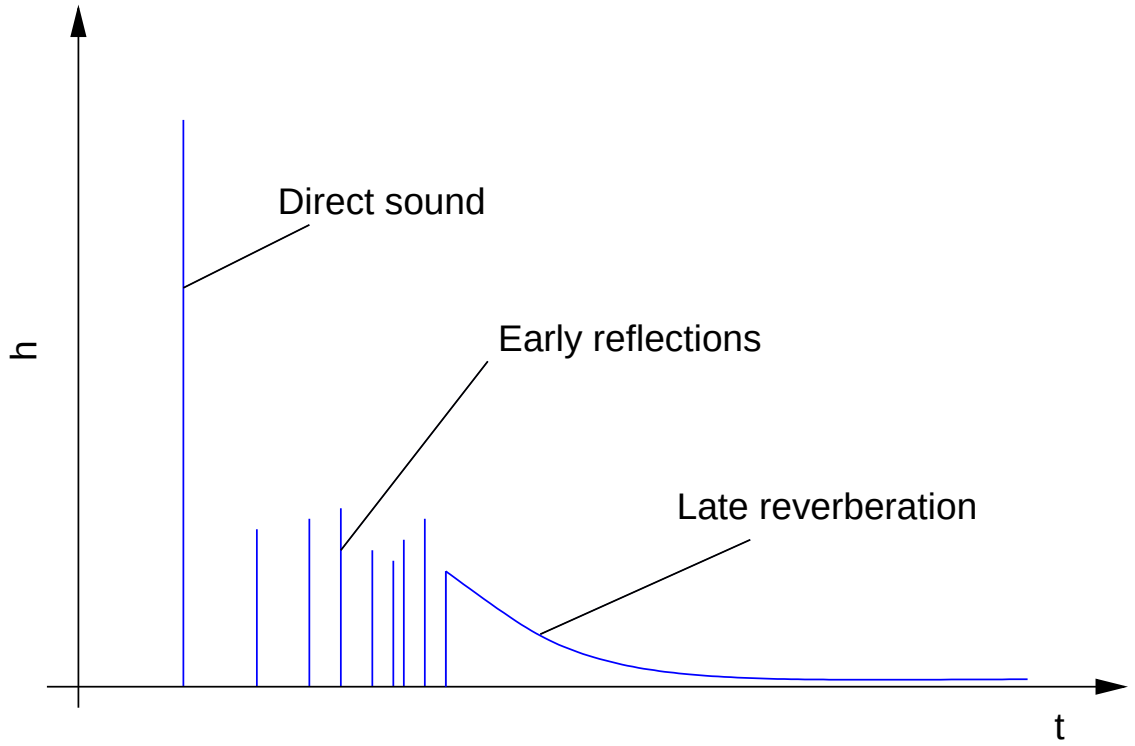
- Echo cancellation:
  $$(\mathbf{W_{yu}} + \mathbf{W_{yx}} * \mathbf{H_{xv}} * \mathbf{W_{vu}}) * \mathbf{u} = \mathbf{0}$$
- Source separation and **dereverberation**:
  $$\mathbf{W_{yx}} * \mathbf{H_{xs}} * \mathbf{s} = \mathbf{s} * \delta(k - k_0)$$
- Noise and interference suppression:
  $$\mathbf{W_{yx}} * \mathbf{n_x} = \mathbf{0}$$

Components of x, i.e., $\mathbf{H_{xs}} * \mathbf{s}$, $\mathbf{H_{xv}} * \mathbf{v}$, $\mathbf{n_x}$, must be separated by **W**!

Fundamentals - Room Impulse Response (RIR) properties

Elements of $\mathbf{H_{wv}}, \mathbf{H_{xv}}, \mathbf{H_{xs}}$ are room impulse responses (RIRs).

Typical structure of RIRs:



Main characteristic parameters:

T_{60}: Time for exponential decay of envelope by 60dB

DRR: Direct-to-Reverberant (Energy) Ratio

Fundamentals - Room Impulse Response (RIR) properties

- **Reverberation time T_{60}**
  - car $\approx$ 50ms
  - concert halls $\approx$ 1 . . . 2s

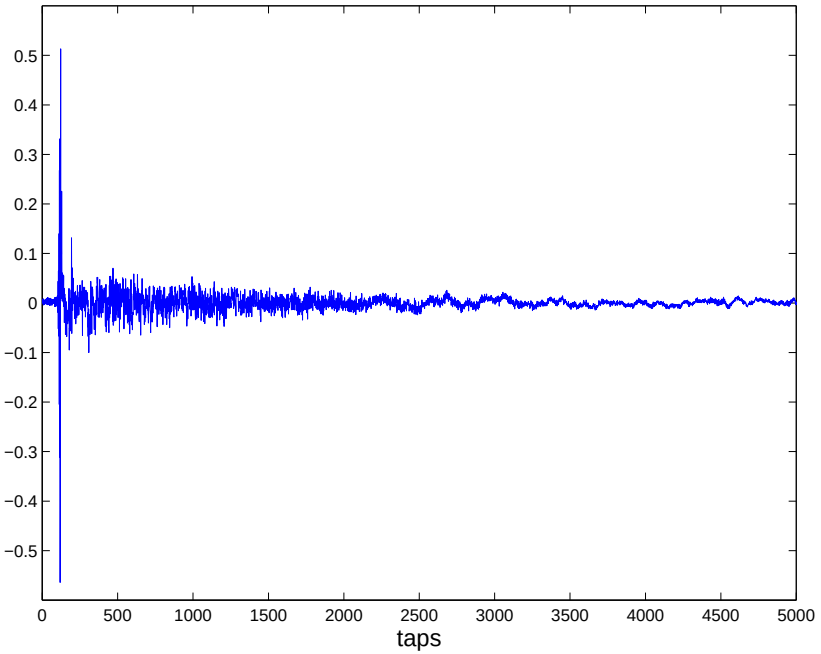
Fundamentals - Room Impulse Response (RIR) properties

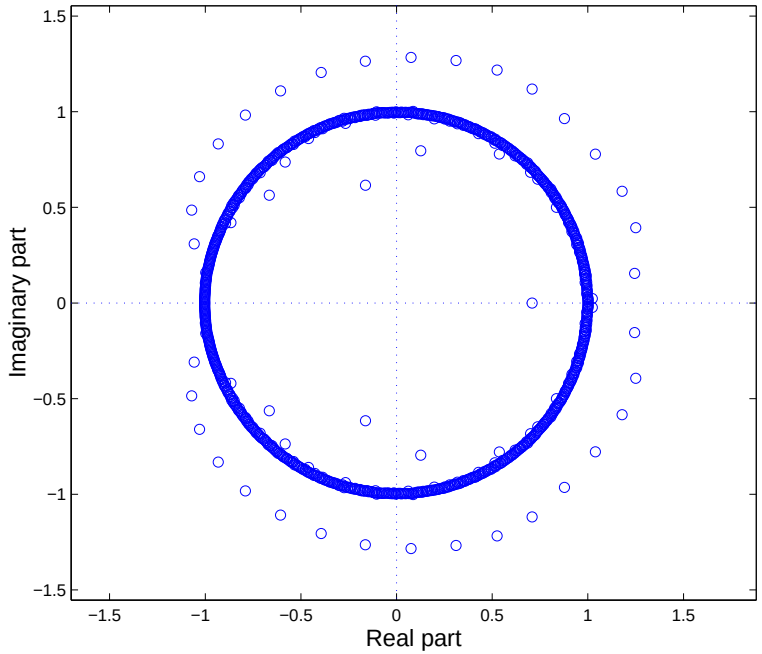
- **Reverberation time T_{60}**
  - car $\approx$ 50ms
  - concert halls $\approx 1 \ldots 2$s
- **FIR models**
  - typically $L_H \approx T_{60} \cdot f_s/3$ coefficients
  - nonminimum-phase
  - many zeros close to unit circle

Fundamentals - Room Impulse Response (RIR) properties

- **Reverberation time T_{60}**
  - car $\approx$ 50ms
  - concert halls $\approx 1 \ldots 2$s
- **FIR models**
  - typically $L_H \approx T_{60} \cdot f_s/3$ coefficients
  - nonminimum-phase
  - many zeros close to unit circle

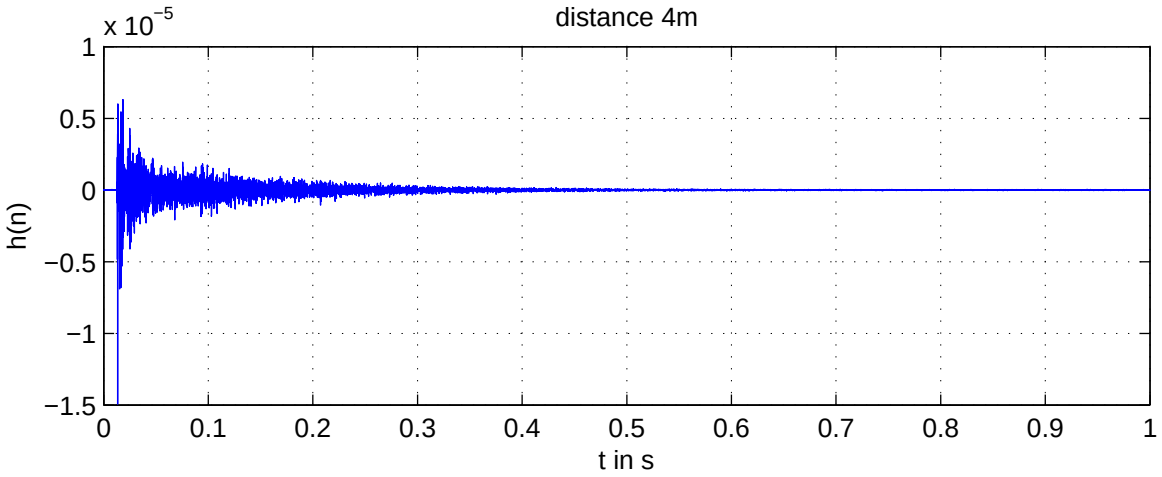
Example: Office 5.5m $\times$ 3m $\times$ 2.8m, $T_{60} \approx$ 300msec, $f_s = 12$kHz.

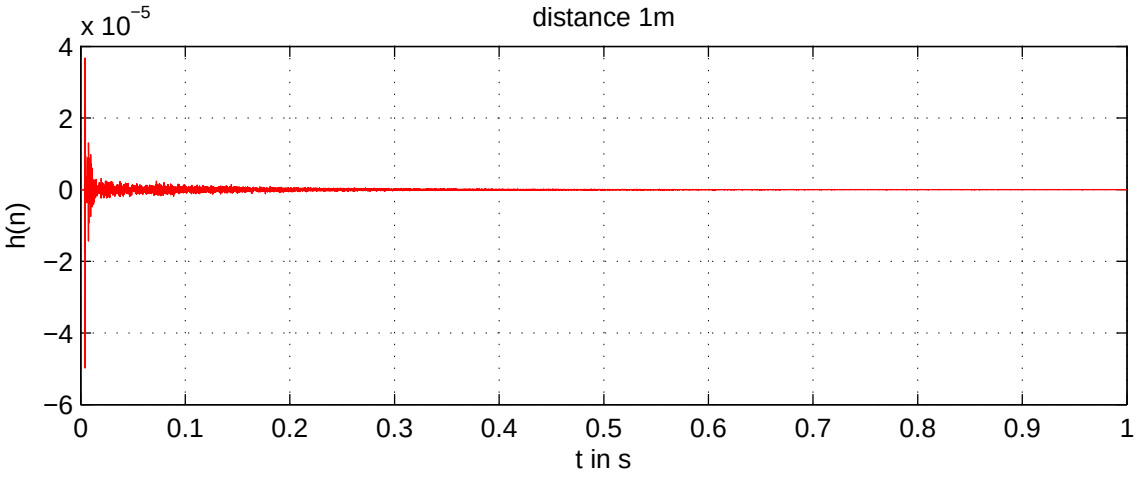




Fundamentals - RIR properties (cont'd)

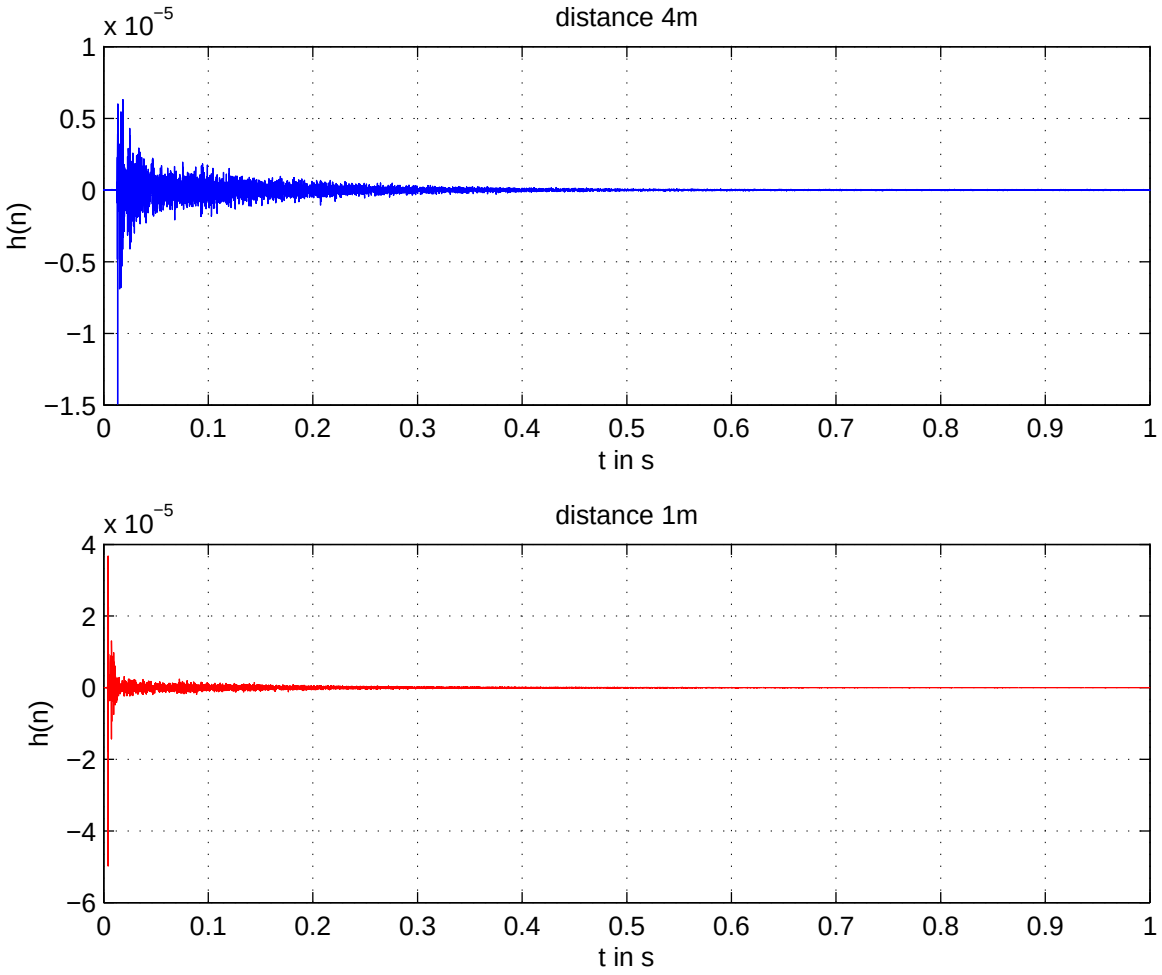
RIRs for varying source-mic distance ($d_1 = 1m$ vs. $d_2 = 4m$, $T_{60} \approx 900ms$)





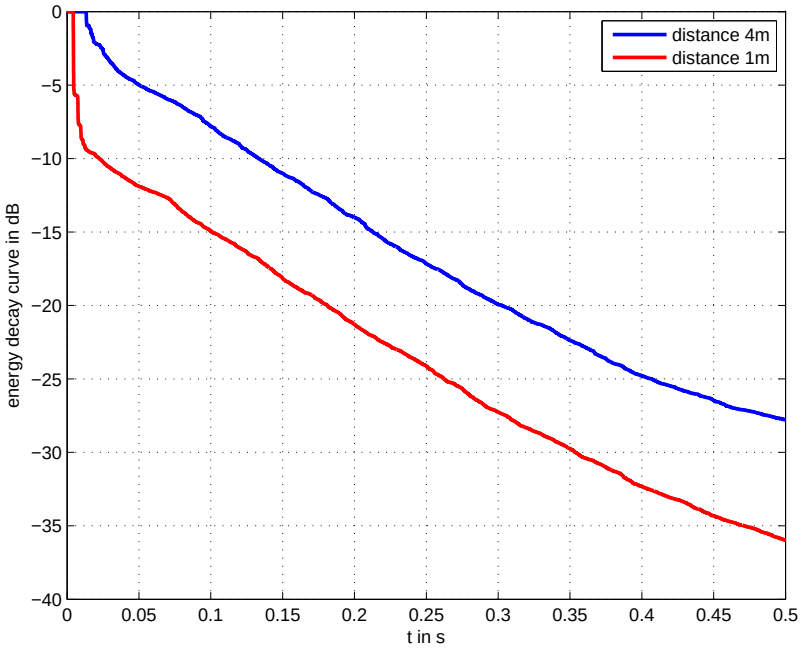
Fundamentals - RIR properties (cont'd)

RIRs for varying source-mic distance ($d_1 = 1m$ vs. $d_2 = 4m$, $T_{60} \approx 900ms$)



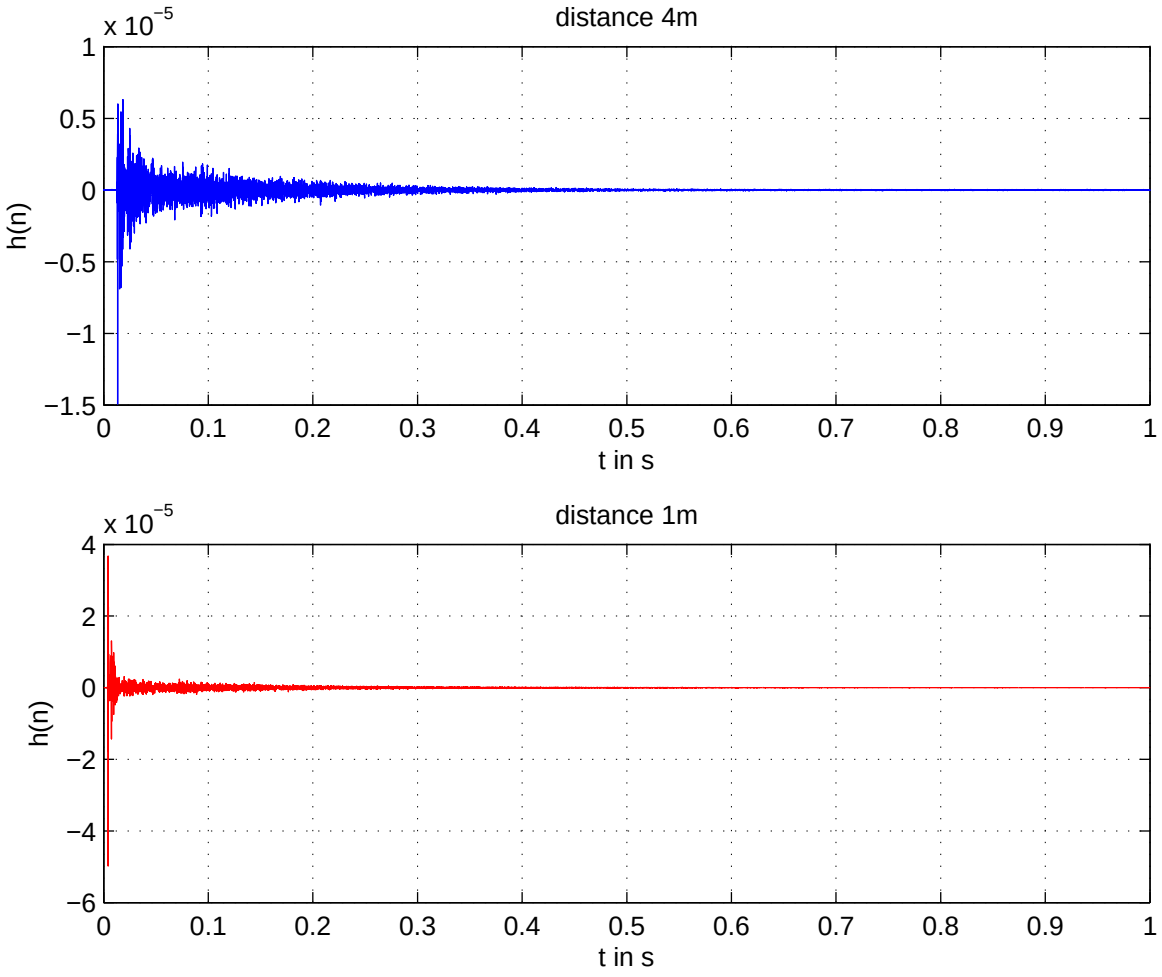
Energy decay curves

(EDC [Schröder 1965])



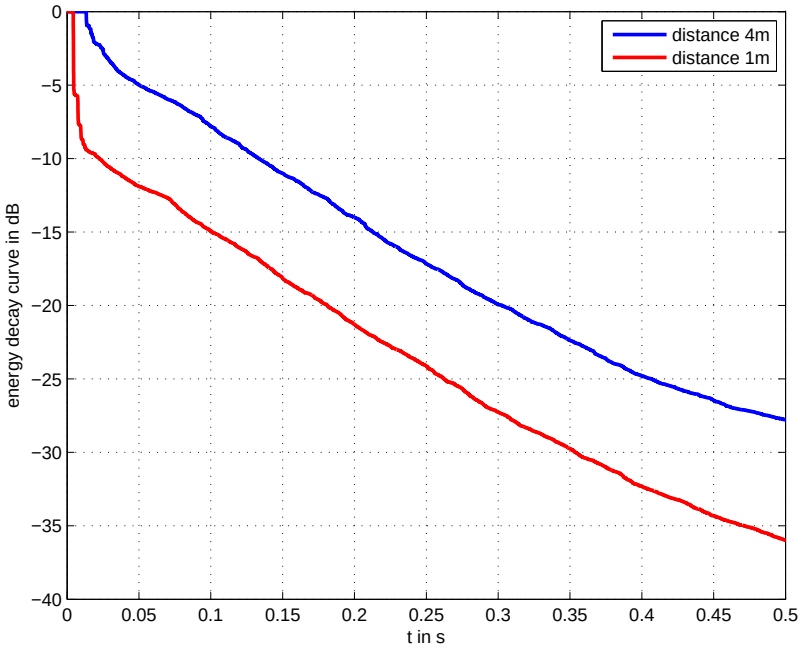
Fundamentals - RIR properties (cont'd)

RIRs for varying source-mic distance ($d_1 = 1m$ vs. $d_2 = 4m$, $T_{60} \approx 900ms$)



Energy decay curves

(EDC [Schröder 1965])



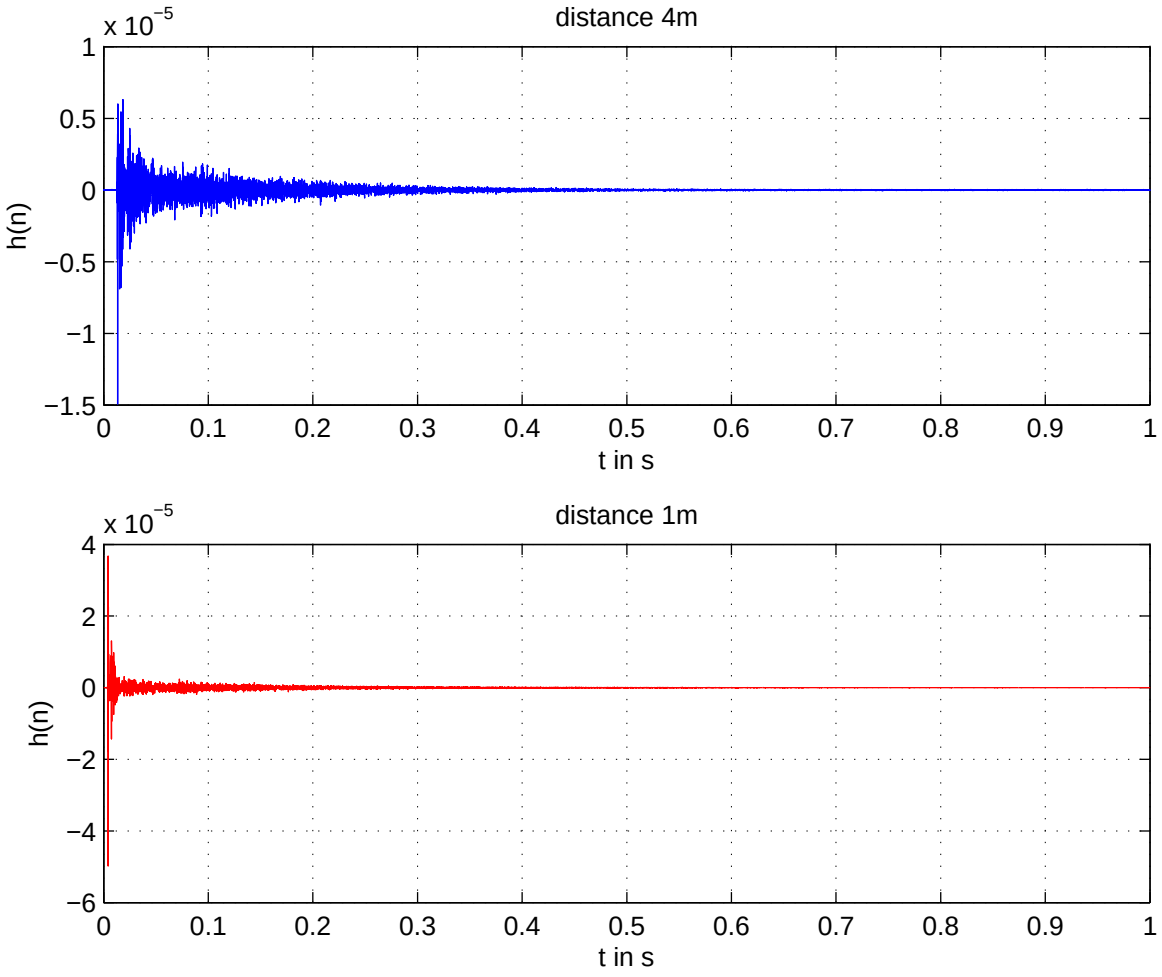
DRR

(Direct-to-Reverberant Energy Ratio)

4.9dB/-4.0dB

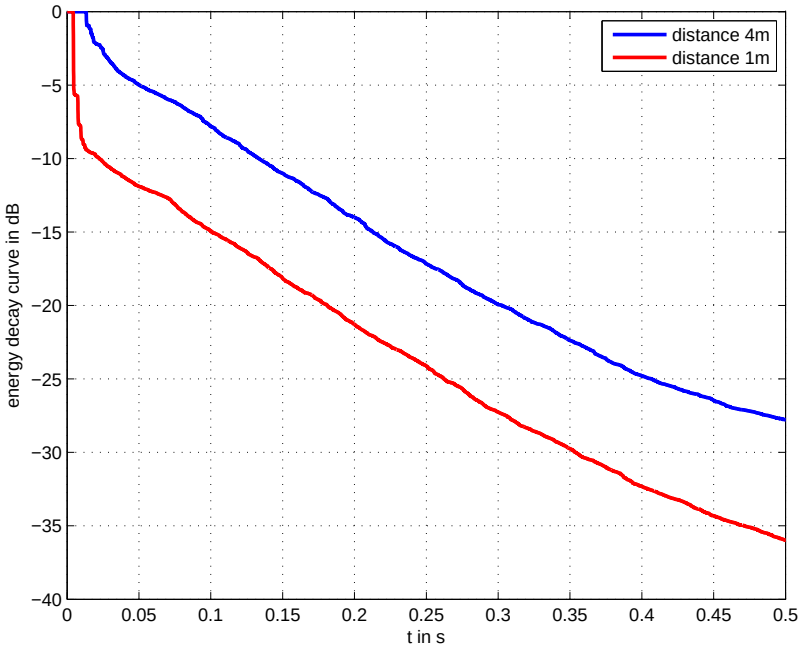
Fundamentals - RIR properties (cont'd)

RIRs for varying source-mic distance ($d_1 = 1m$ vs. $d_2 = 4m$, $T_{60} \approx 900ms$)



Energy decay curves

(EDC [Schröder 1965])



DRR

(Direct-to-Reverberant Energy Ratio)

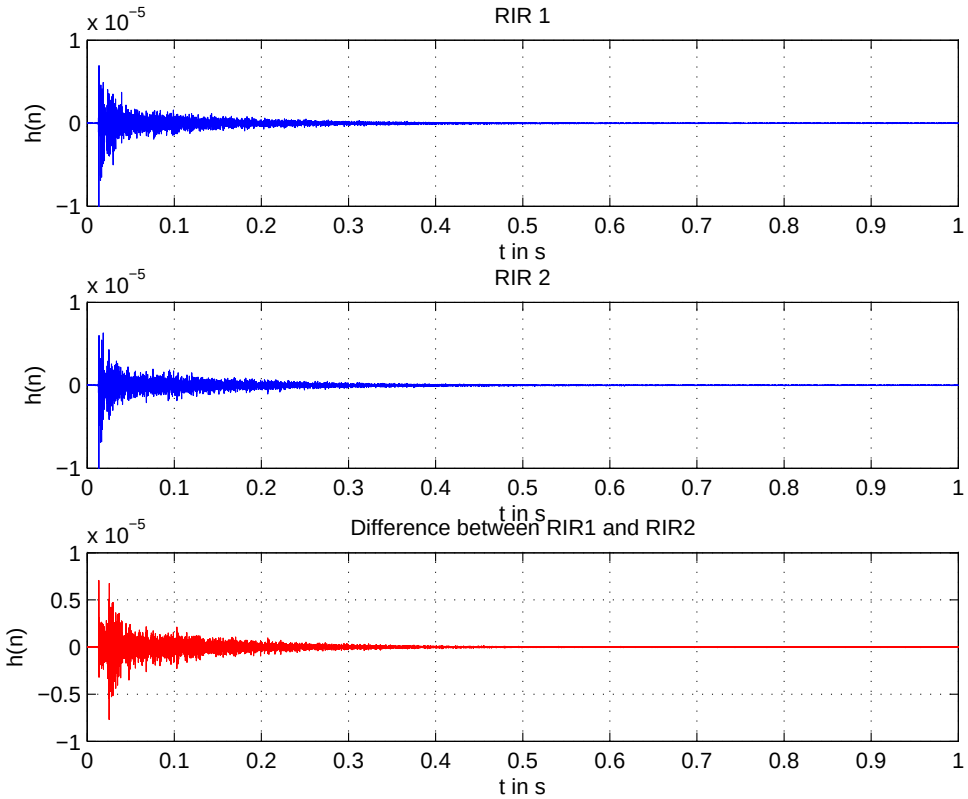
4.9dB/-4.0dB

RIR, DRR $\Leftrightarrow$ Reverberation time T_{60}

Fundamentals - RIR properties (cont'd)

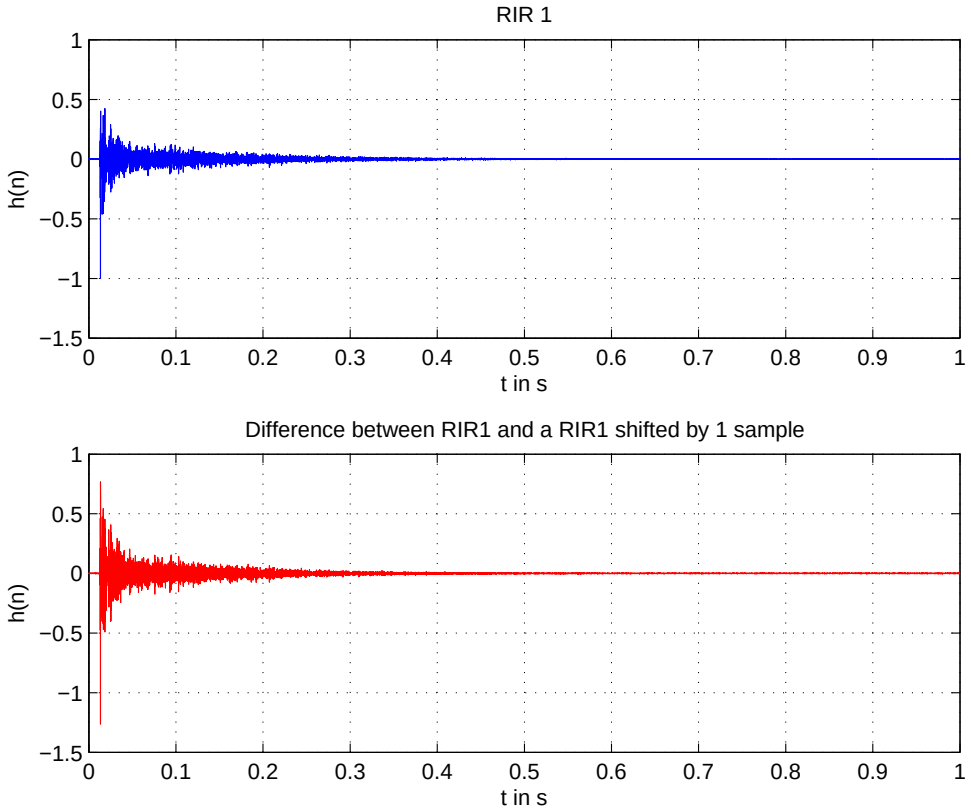
Variability with displacements:

Mic displacement 4.2cm
(source distance d=4m):



System error norm: 0.23dB

Shift of RIR by 1 sample:

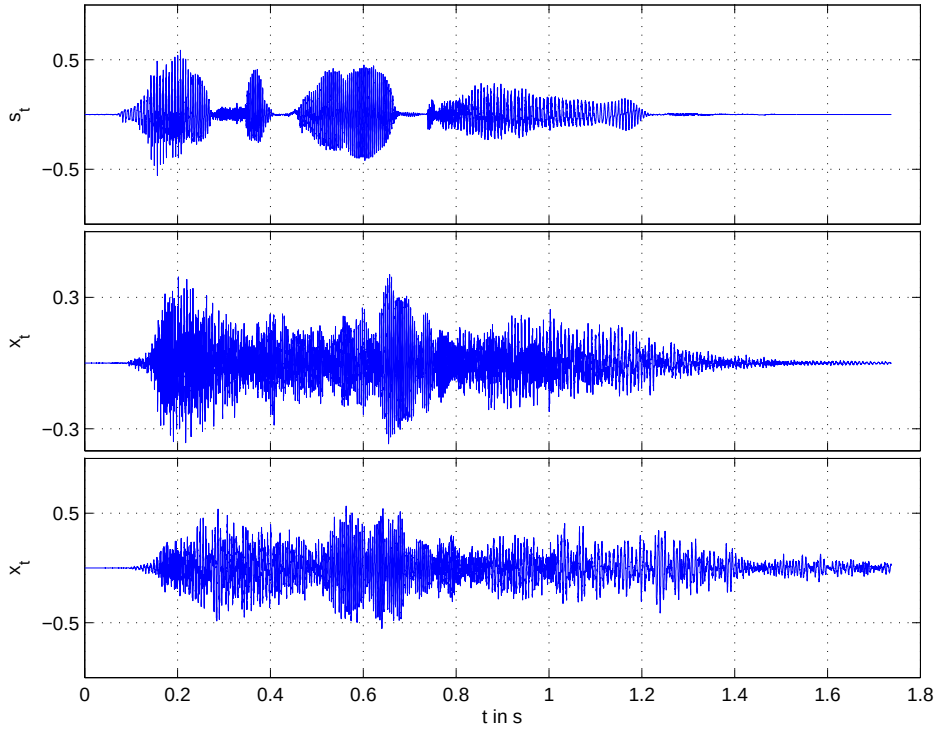


System error norm: 2.56dB

Fundamentals - Reverberation in signal representations

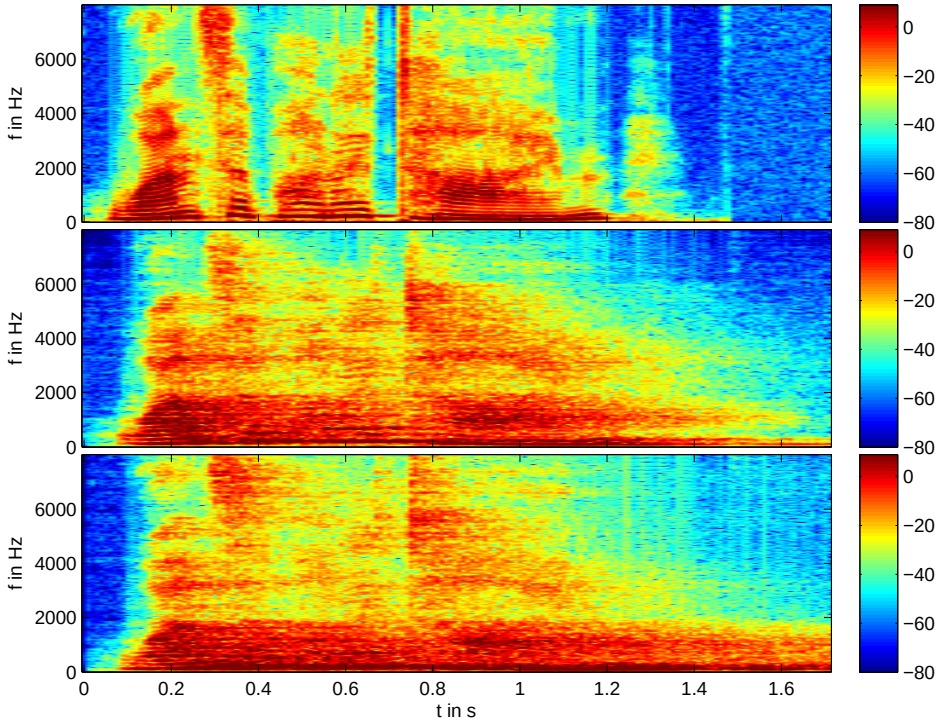
Clean vs. reverberated with $T_{60} \approx 900ms$, $d_1 = 4m$ and $T_{60} \approx 3.1s$, $d_2 = 5m$

Time-domain:



Pauses filled!

STFT domain:

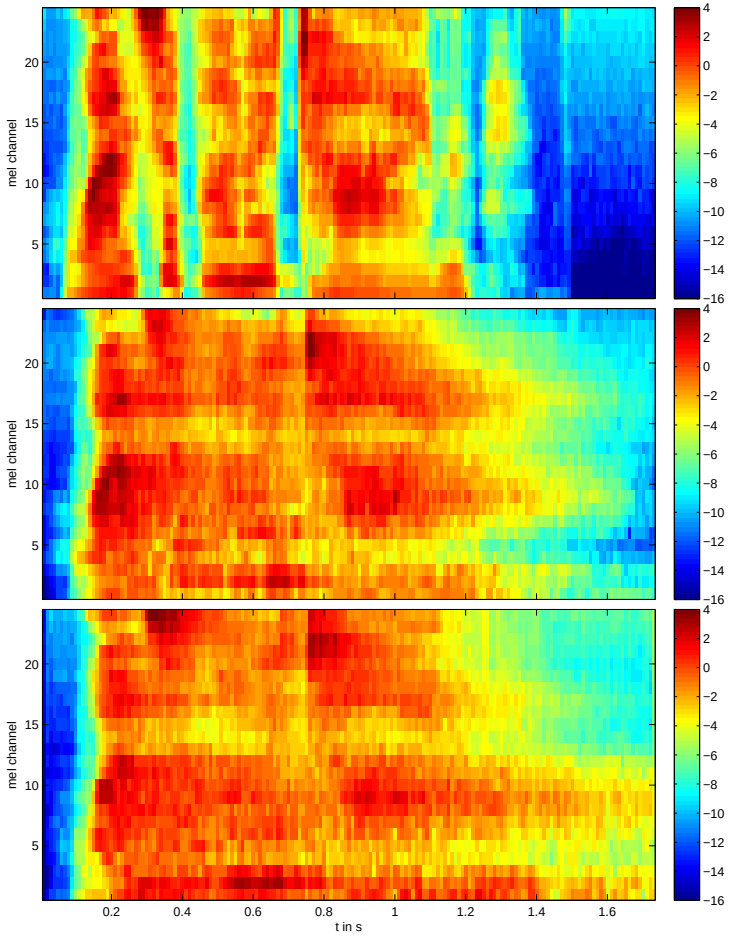


Pauses filled!

Fundamentals - Reverberation in ASR features

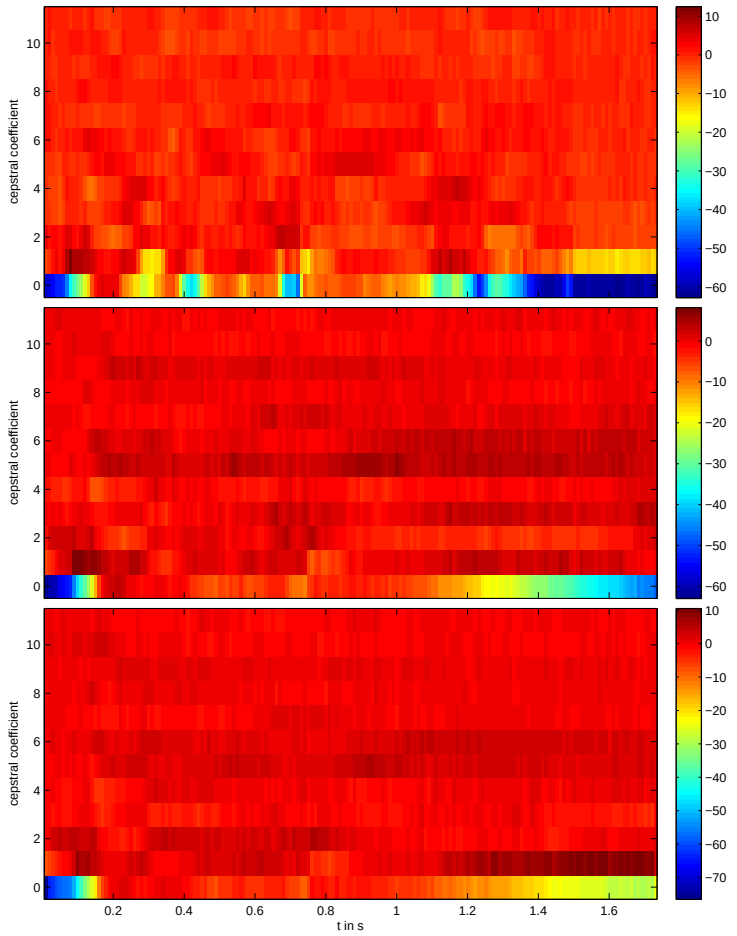
Clean vs. reverberated with $T_{60} \approx 900ms$, $d_1 = 4m$ and $T_{60} \approx 3.1s$, $d_2 = 5m$

Logmelspec domain:



Pauses filled!

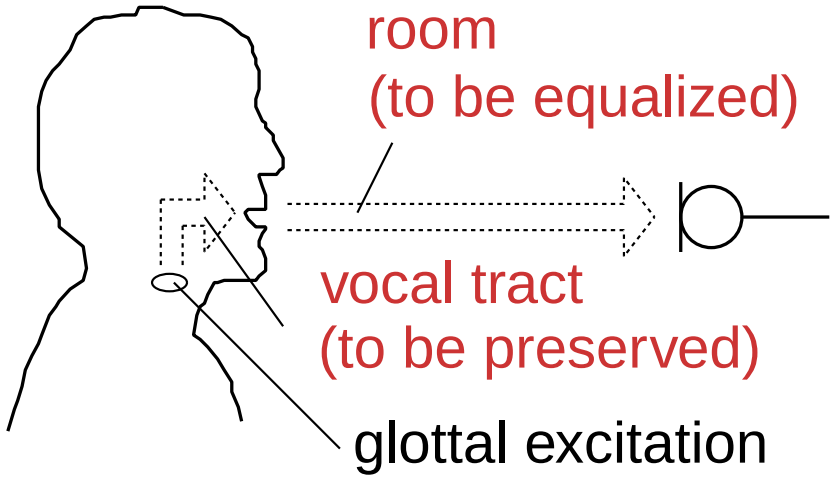
MFCC domain:



Pauses of c_0 filled!

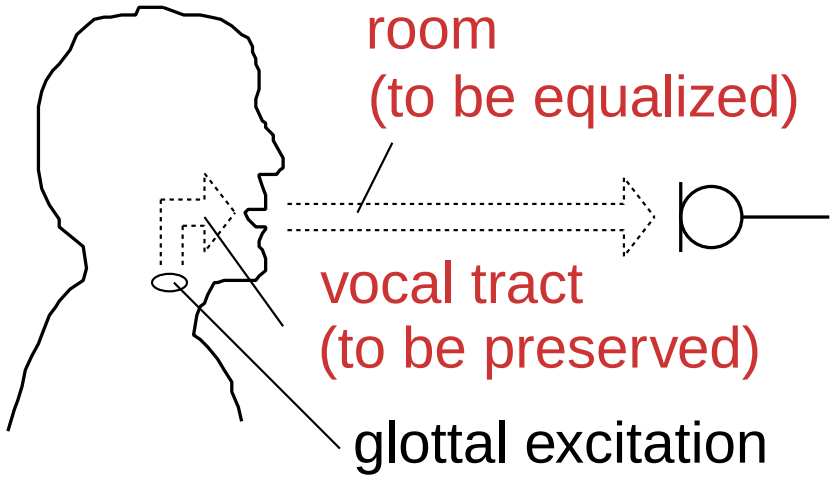
Dereverberation for Speech Enhancement

Basic Idea: Separate speech production from RIR, equalize the latter



Dereverberation for Speech Enhancement

Basic Idea: Separate speech production from RIR, equalize the latter

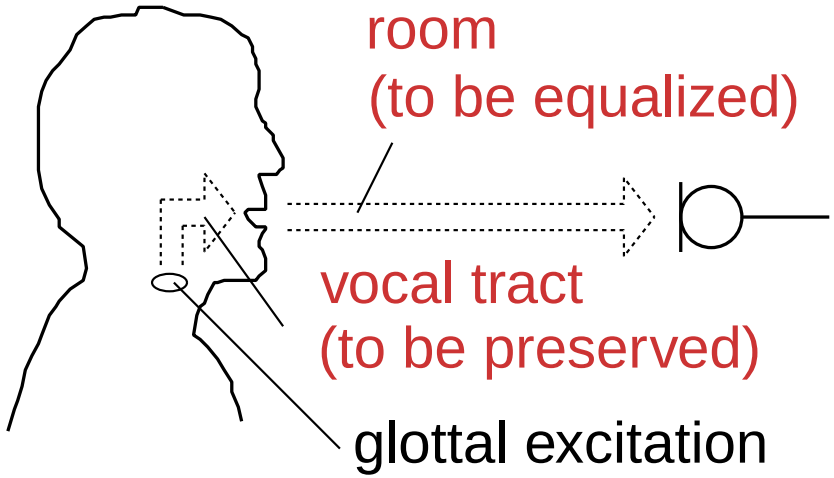


'Blind' problem!

(no reference signal for RIR input)

Dereverberation for Speech Enhancement

Basic Idea: Separate speech production from RIR, equalize the latter



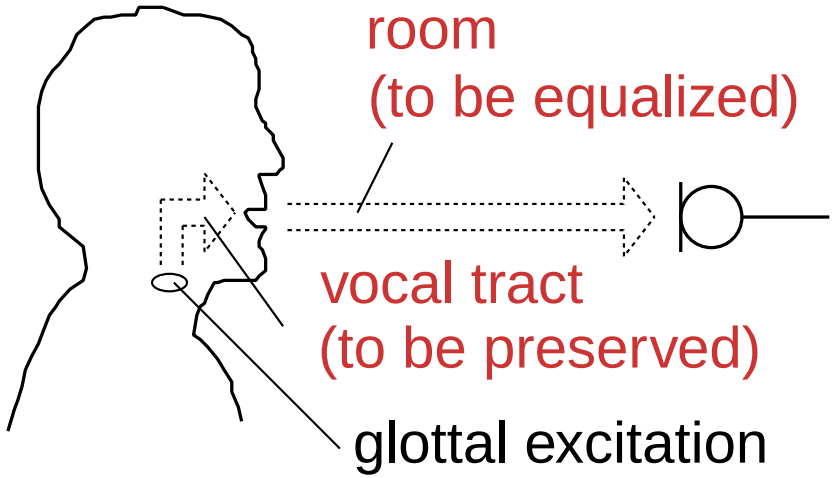
Distinction:

- **Partial Deconvolution**
  (removes reverberation by RIR inversion, ideally without speech distortion)

'Blind' problem!
(no reference signal for RIR input)

Dereverberation for Speech Enhancement

Basic Idea: Separate speech production from RIR, equalize the latter



'Blind' problem!
(no reference signal for RIR input)

Distinction:

- **Partial Deconvolution**
  (removes reverberation by RIR inversion, ideally without speech distortion)

⇕

- **Reverberation Suppression**
  (compromise between dereverberation and signal distortion necessary)

Dereverberation for Signal Enhancement (cont'd)

Dealing with 'Blindness' by exploiting

Prior Knowledge on

A, Speech production models (e.g., source-filter model, HMM) and signal properties (nonwhiteness, nonstationarity, nongaussianity)

B, Room acoustics parameter (e.g., T_{60})

C, Location and radiation characteristics of speech source

Dereverberation for Signal Enhancement (cont'd)

Dealing with 'Blindness' by exploiting

Prior Knowledge on

A, Speech production models (e.g., source-filter model, HMM) and signal properties (nonwhiteness, nonstationarity, nongaussianity)

B, Room acoustics parameter (e.g., T_{60})

C, Location and radiation characteristics of speech source

and some **Useful Assumptions**

D, Joint moments (e.g., correlation) of signal samples:
Small lags characterize speech $\Leftrightarrow$ Large lags characterize reverberation

Dereverberation for Signal Enhancement (cont'd)

Dealing with 'Blindness' by exploiting

Prior Knowledge on

A, Speech production models (e.g., source-filter model, HMM) and signal properties (nonwhiteness, nonstationarity, nongaussianity)

B, Room acoustics parameter (e.g., T_{60})

C, Location and radiation characteristics of speech source

and some **Useful Assumptions**

D, Joint moments (e.g., correlation) of signal samples:
Small lags characterize speech $\Leftrightarrow$ Large lags characterize reverberation

E, Speech signal statistics change faster than RIRs

Dereverberation for Signal Enhancement (cont'd)

Dealing with 'Blindness' by exploiting

Prior Knowledge on

A, Speech production models (e.g., source-filter model, HMM) and signal properties (nonwhiteness, nonstationarity, nongaussianity)

B, Room acoustics parameter (e.g., T_{60})

C, Location and radiation characteristics of speech source

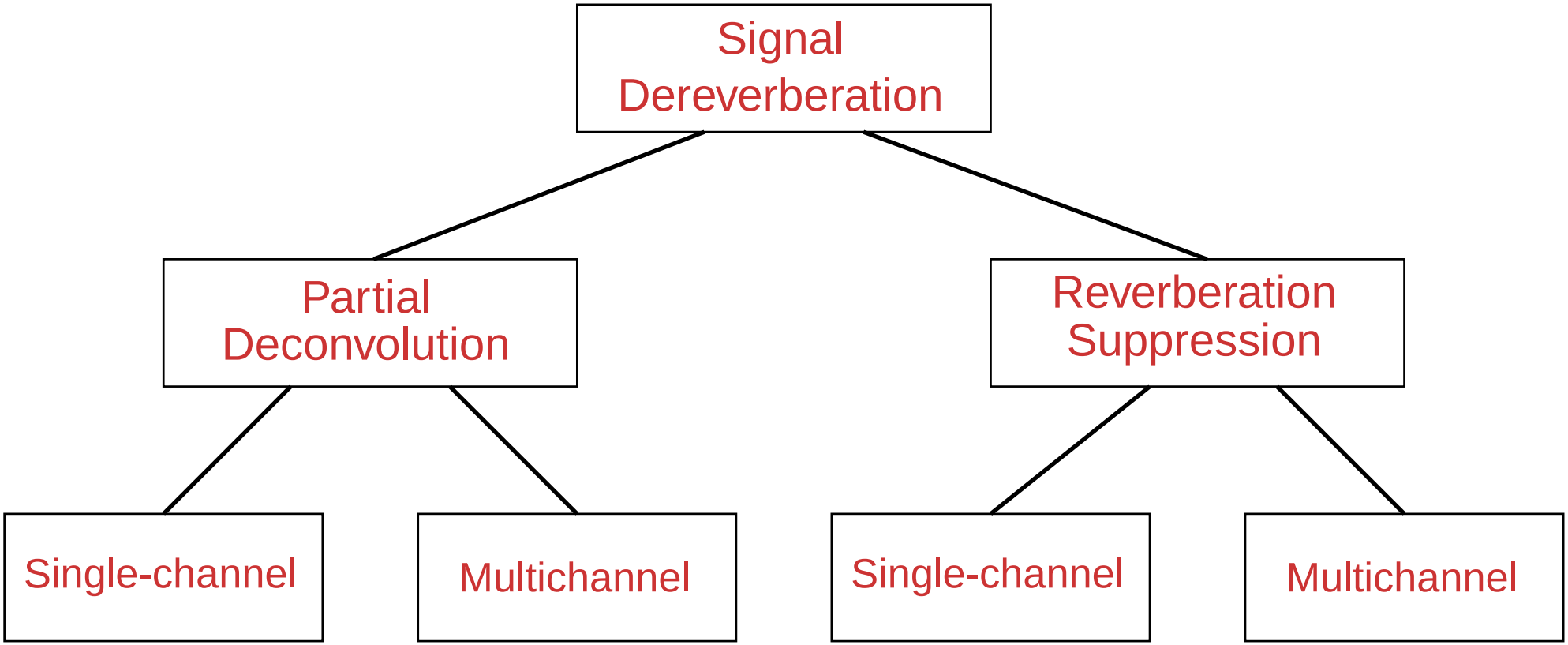
and some **Useful Assumptions**

D, Joint moments (e.g., correlation) of signal samples:
Small lags characterize speech $\Leftrightarrow$ Large lags characterize reverberation

E, Speech signal statistics change faster than RIRs

F, Multichannel recordings: Speech component is the same $\Leftrightarrow$ RIRs are different

Dereverberation for Signal Enhancement - Approaches



Dereverberation - Single-Channel Partial Deconvolution

Single-channel partial deconvolution

- Can exploit speech models and properties (A) and correlation and stationarity assumptions (D, E) for identifying RIR estimate

Dereverberation - Single-Channel Partial Deconvolution

Single-channel partial deconvolution

- Can exploit speech models and properties (A) and correlation and stationarity assumptions (D, E) for identifying RIR estimate
- Inversion of a single RIR involves [Neely 1979]
  - removing the allpass component of the nonminimum-phase RIR $\rightarrow$ approximated by delay
  - inverting zeros close to, or on unit circle $\rightarrow$ approximation by 'channel shortening'

Dereverberation - Single-Channel Partial Deconvolution

Single-channel partial deconvolution

- Can exploit speech models and properties (A) and correlation and stationarity assumptions (D, E) for identifying RIR estimate
- Inversion of a single RIR involves [Neely 1979]
  - removing the allpass component of the nonminimum-phase RIR $\rightarrow$ approximated by delay
  - inverting zeros close to, or on unit circle $\rightarrow$ approximation by 'channel shortening'
- for realization problems see, e.g., [Morjopoulos 1994], [Naylor 2010]

Dereverberation - Multichannel Partial Deconvolution

Multichannel partial deconvolution

Dereverberation - Multichannel Partial Deconvolution

Multichannel partial deconvolution

- Can additionally exploit spatial diversity (incl. assumption F) and prior knowledge of source location and radiation characteristic (C) for identifying RIR
- Spatial diversity facilitates RIR identification

Dereverberation - Multichannel Partial Deconvolution

Multichannel partial deconvolution

- Can additionally exploit spatial diversity (incl. assumption F) and prior knowledge of source location and radiation characteristic (C) for identifying RIR
- Spatial diversity facilitates RIR identification
- Perfect inversion with FIR filters is possible (MINT [Miyoshi 1988])
  - exact knowledge of RIR lengths required
  - no common zeros of RIRs allowed

Dereverberation - Multichannel Partial Deconvolution

Multichannel partial deconvolution

- Can additionally exploit spatial diversity (incl. assumption F) and prior knowledge of source location and radiation characteristic (C) for identifying RIR
- Spatial diversity facilitates RIR identification
- Perfect inversion with FIR filters is possible (MINT [Miyoshi 1988])
  - exact knowledge of RIR lengths required
  - no common zeros of RIRs allowed
- Indirect approaches often invert in subbands for robustness (e.g. [Naylor 2005])

Dereverberation - Multichannel Partial Deconvolution

Multichannel partial deconvolution

- Can additionally exploit spatial diversity (incl. assumption F) and prior knowledge of source location and radiation characteristic (C) for identifying RIR
- Spatial diversity facilitates RIR identification
- Perfect inversion with FIR filters is possible (MINT [Miyoshi 1988])
  - exact knowledge of RIR lengths required
  - no common zeros of RIRs allowed
- Indirect approaches often invert in subbands for robustness (e.g. [Naylor 2005])
- Direct approaches to identify a robust inverse exist (e.g. [Buchner 2004], [Buchner 2010], and below!)

Dereverberation - Single-Channel Reverberation Suppression

Single-channel Reverberation Suppression

Dereverberation - Single-Channel Reverberation Suppression

Single-channel Reverberation Suppression

- can exploit speech models and properties (A) and correlation and stationarity assumptions (D, E), e.g., for
  - equalizing the vocal tract IR and suppressing reverberation in the LPC residual (e.g., [Yegnanarayana 2000], [Gaubitch 2006])

Dereverberation - Single-Channel Reverberation Suppression

Single-channel Reverberation Suppression

- can exploit speech models and properties (A) and correlation and stationarity assumptions (D, E), e.g., for
  - equalizing the vocal tract IR and suppressing reverberation in the LPC residual (e.g., [Yegnanarayana 2000], [Gaubitch 2006])
- can exploit prior knowledge on room acoustics (e.g., T_{60}) to estimate PSD of reverberation and use spectral subtraction methods as common for additive noise (e.g., [Lebart 2001])

Multichannel Reverberation Suppression

Dereverberation - Multichannel Reverberation Suppression

Multichannel Reverberation Suppression

- can additionally exploit spatial diversity (incl. assumption F) and prior knowledge on source location and radiation characteristic (C), e.g.,

Multichannel Reverberation Suppression

- can additionally exploit spatial diversity (incl. assumption F) and prior knowledge on source location and radiation characteristic (C), e.g.,
  - beamforming using only prior knowledge of source location and radiation characteristic (C) (e.g., [Griebel 2001])

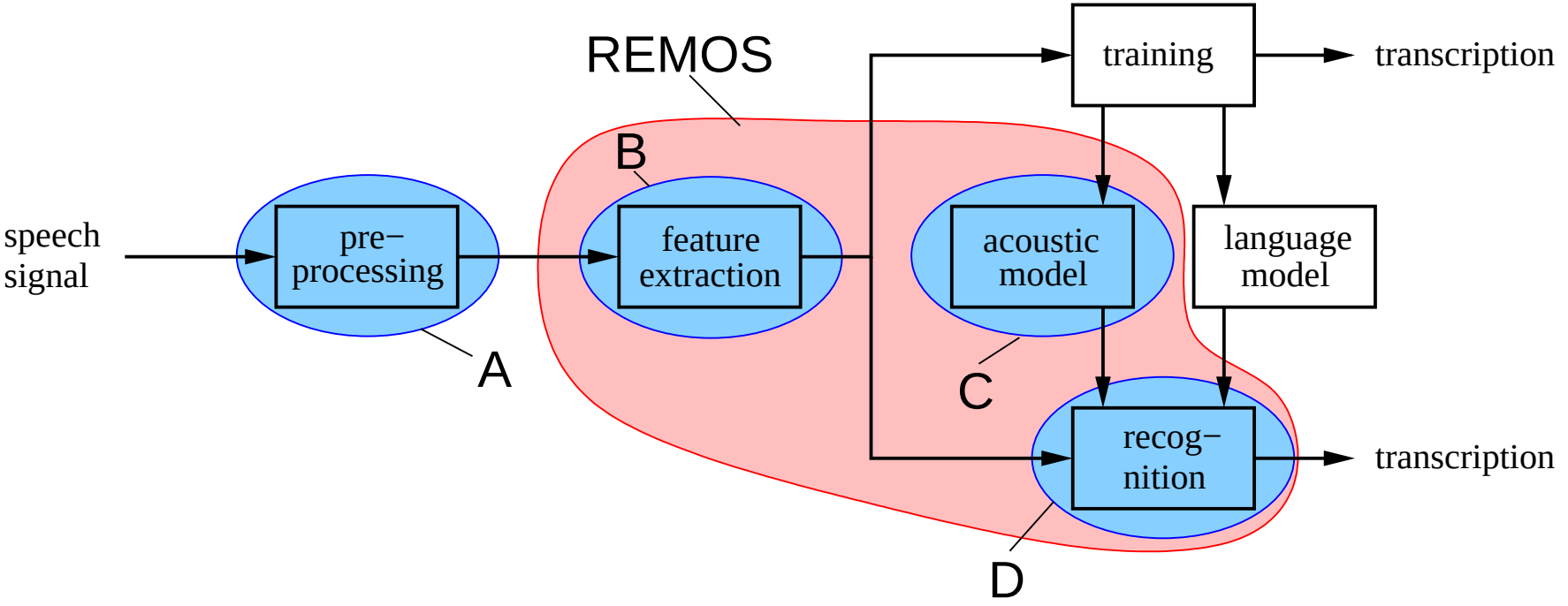
Multichannel Reverberation Suppression

- can additionally exploit spatial diversity (incl. assumption F) and prior knowledge on source location and radiation characteristic (C), e.g.,
  - beamforming using only prior knowledge of source location and radiation characteristic (C) (e.g., [Griebel 2001])
  - spatial diversity for multichannel spectral subtraction (e.g., [Allen 1977]), or subspace methods (e.g., [Gannot 2003])

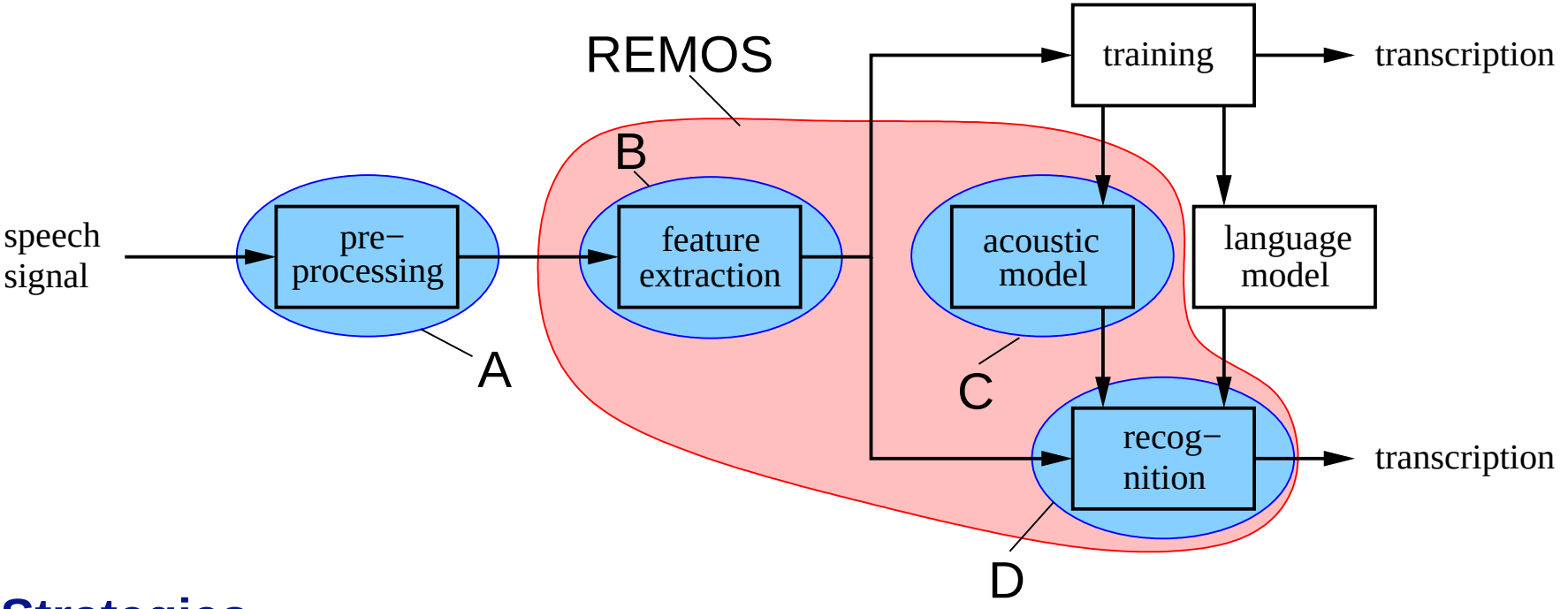
Multichannel Reverberation Suppression

- can additionally exploit spatial diversity (incl. assumption F) and prior knowledge on source location and radiation characteristic (C), e.g.,
  - beamforming using only prior knowledge of source location and radiation characteristic (C) (e.g., [Griebel 2001])
  - spatial diversity for multichannel spectral subtraction (e.g., [Allen 1977]), or subspace methods (e.g., [Gannot 2003])
  - spatial diversity complemented by prior knowledge on room acoustics parameter (e.g., [Habets 2005])

Block diagram of ASR system



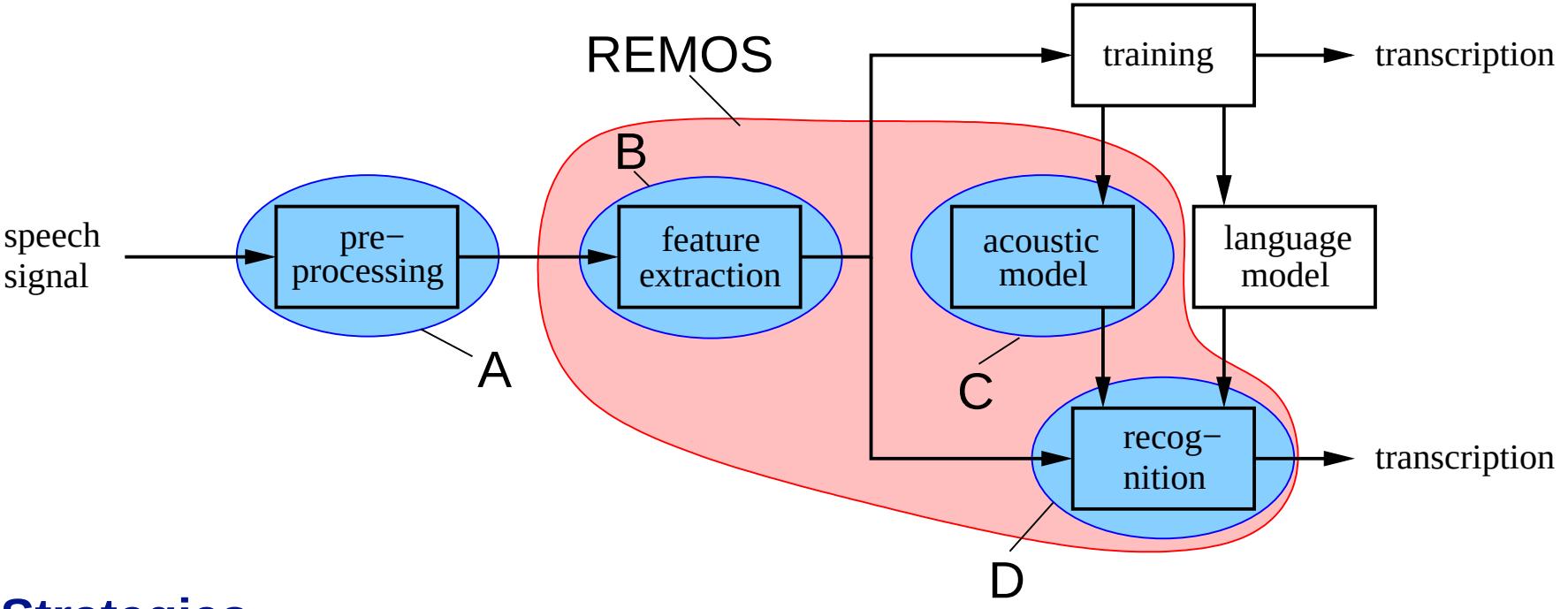
Block diagram of ASR system



Strategies

A) signal-based approaches

Block diagram of ASR system

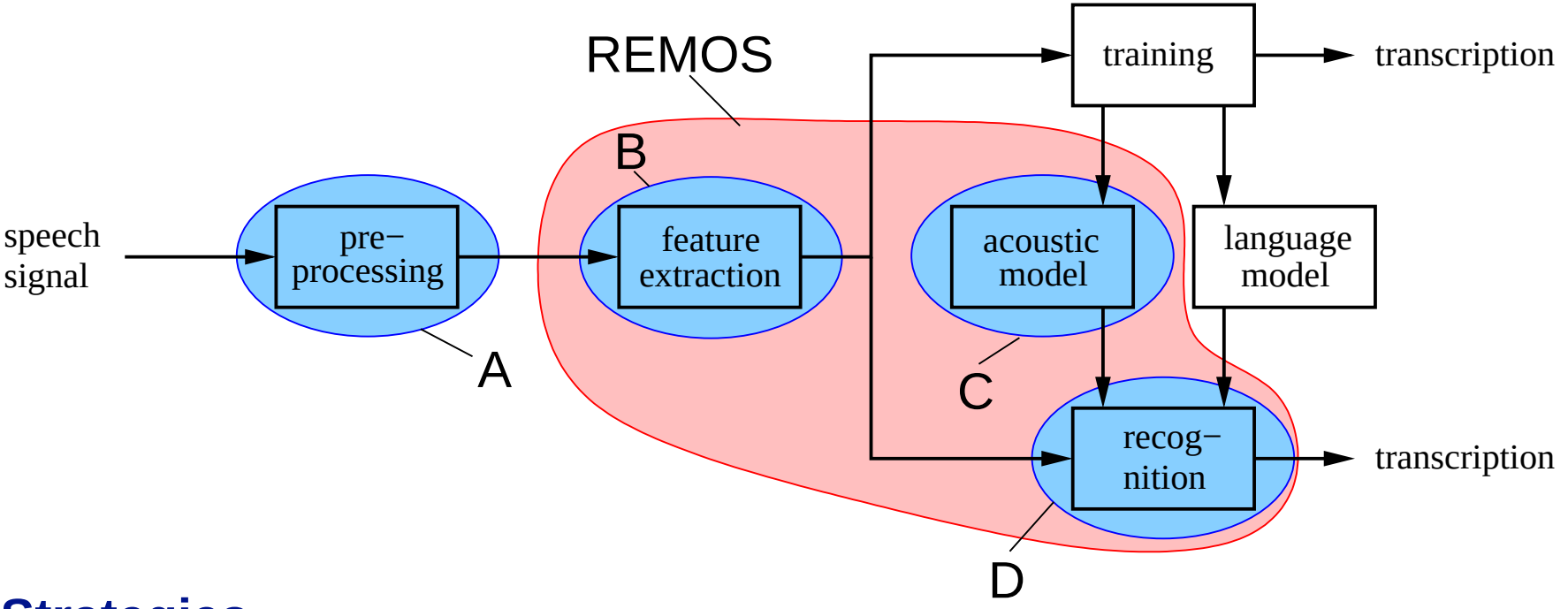


Strategies

A) signal-based approaches

B) feature-based approaches

Block diagram of ASR system



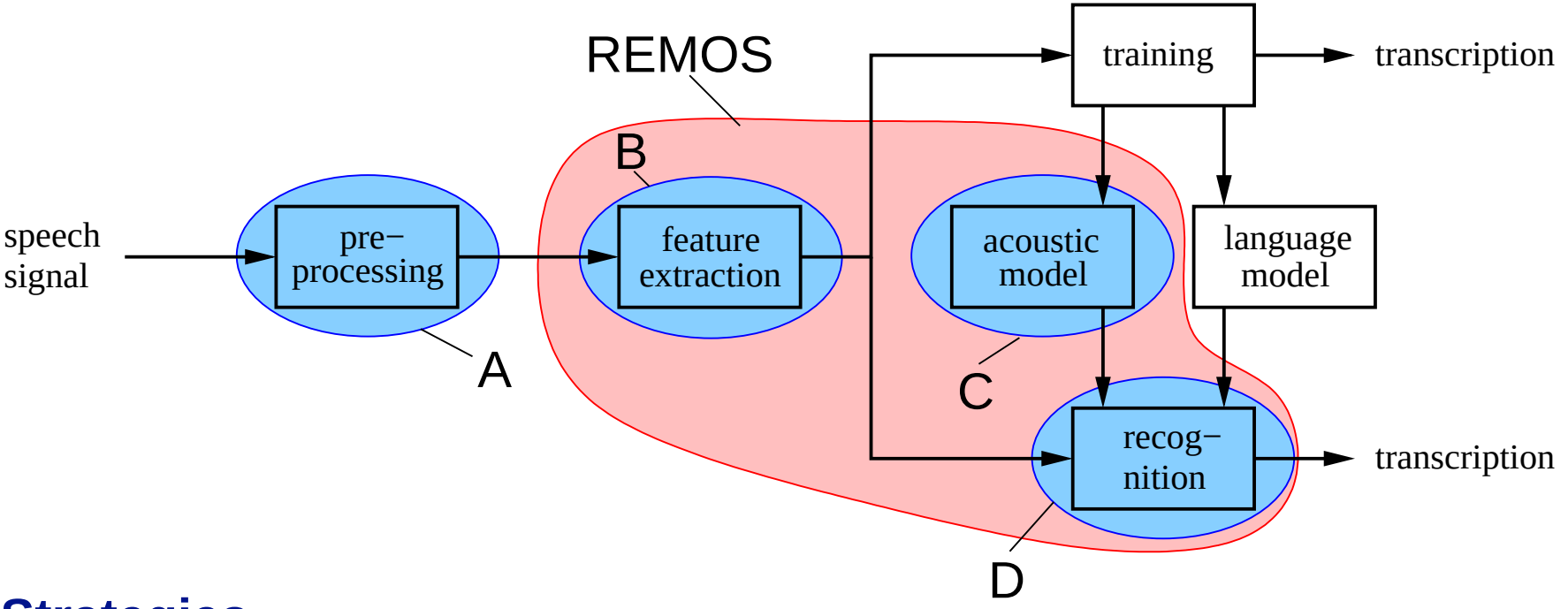
Strategies

A) signal-based approaches

B) feature-based approaches

C) model-based approaches

Block diagram of ASR system



Strategies

A) signal-based approaches

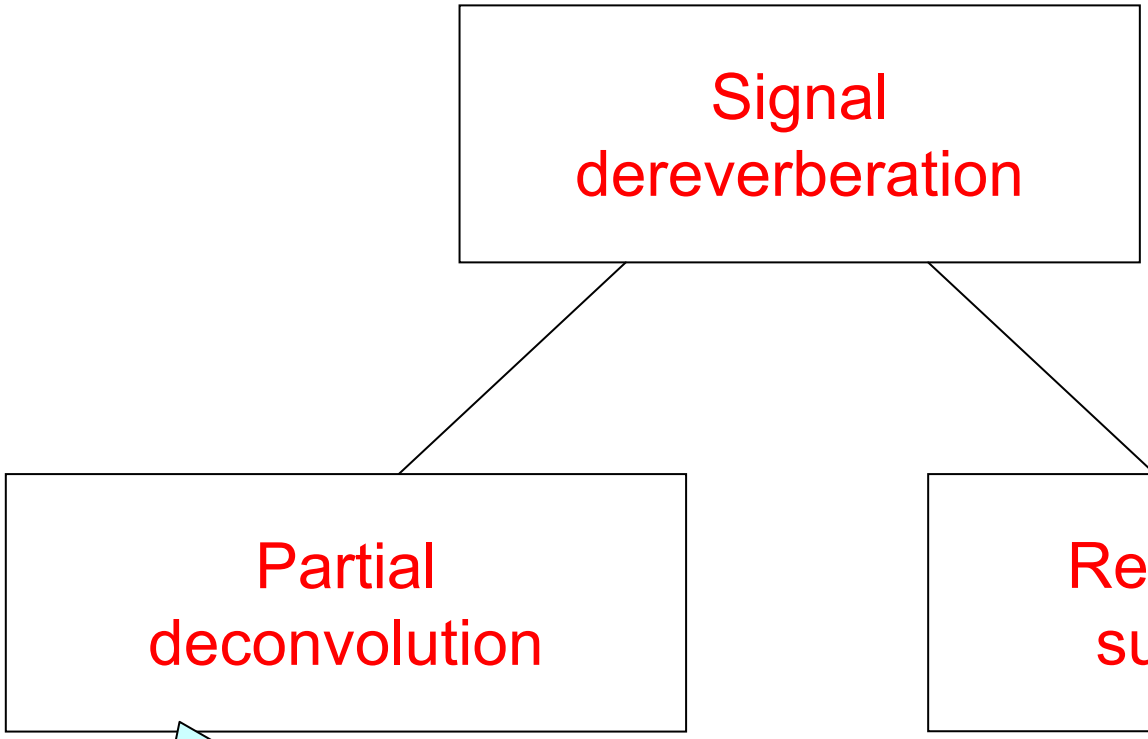
B) feature-based approaches

C) model-based approaches

D) decoder-based approaches

Part II.
Multichannel blind inverse filtering

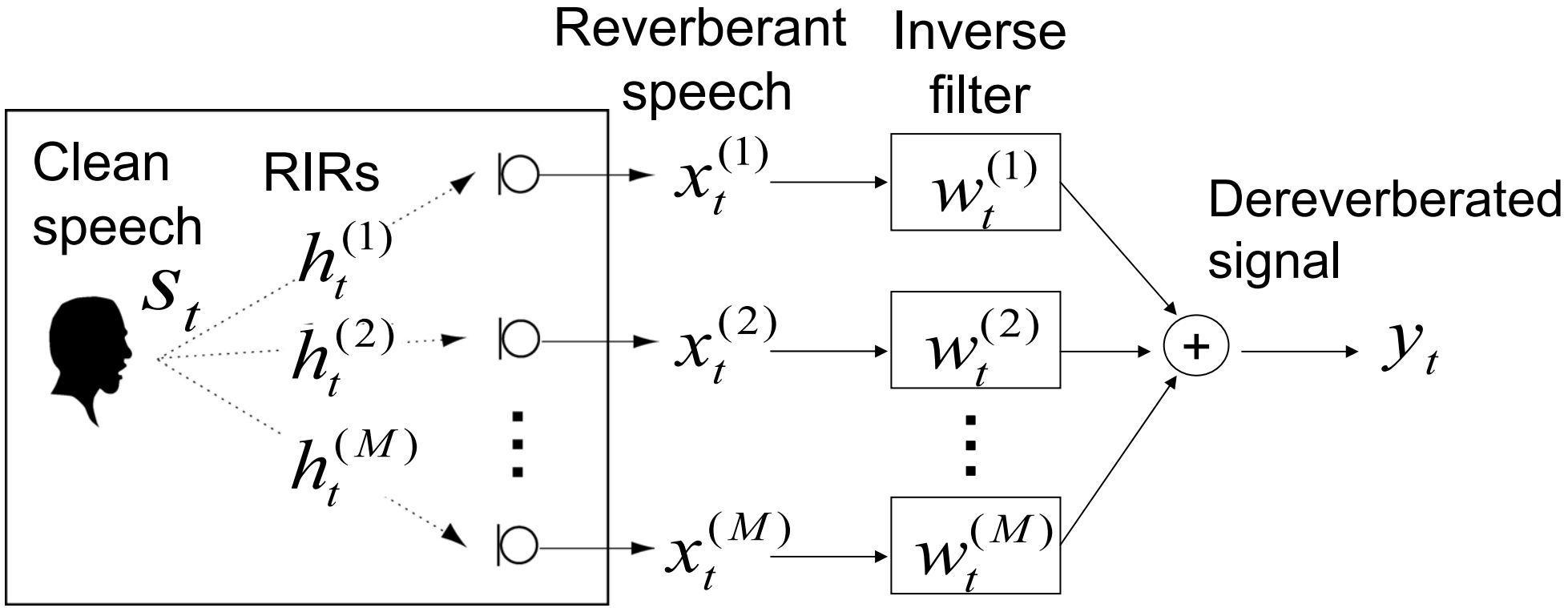
Two approaches for signal dereverberation



"Robust" blind inverse filtering is the main topic of part II

[Lebart 2001], [Habets 2005], [Löllman 2009], [Erkelens 2010], [Kameoka 2009], [Jeub 2010] and others

Multichannel inverse filtering



Linear filtering: $$y_t = \sum_{m=1}^{M} \sum_{\tau=0}^{K} w_\tau^{(m)} x_{t-\tau}^{(m)}$$

Goal: estimate $\{w_t^{(m)}\}$ s.t. $y_t = s_t$

m : mic. index

t : time index

$\{\cdot\}$: a set of variables for all t and m

Part II. Multichannel blind inverse filtering

- **Example applications**
  - **Professional audio post production**
  - **Meeting recognition with microphone arrays**

- Fundamentals: dereverberation with inverse filtering
  - What is inverse filter
  - Robust ‘approximate’ inverse filter

- Blind inverse filtering
  - Overview of basic approaches
  - Closer look: multichannel linear prediction with time-varying source model

- Integration with blind source separation

Application to audio post-production

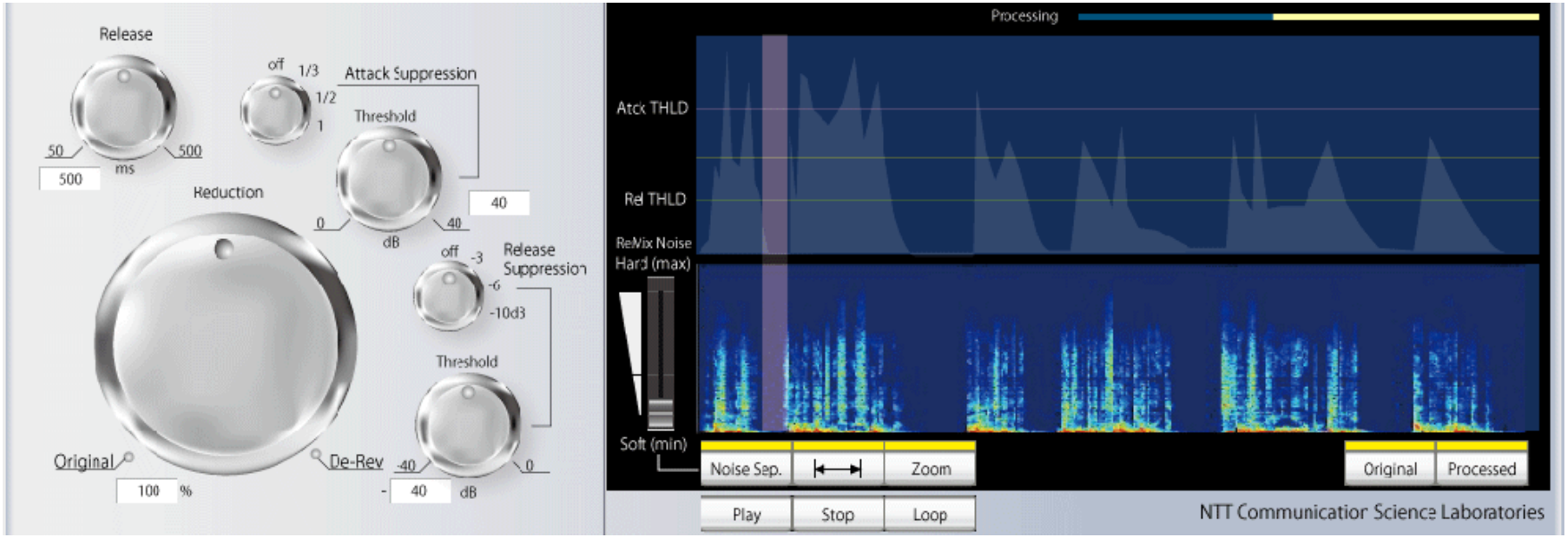
[Movies/TV creation]

Step1:
Sound&video recording (on location)

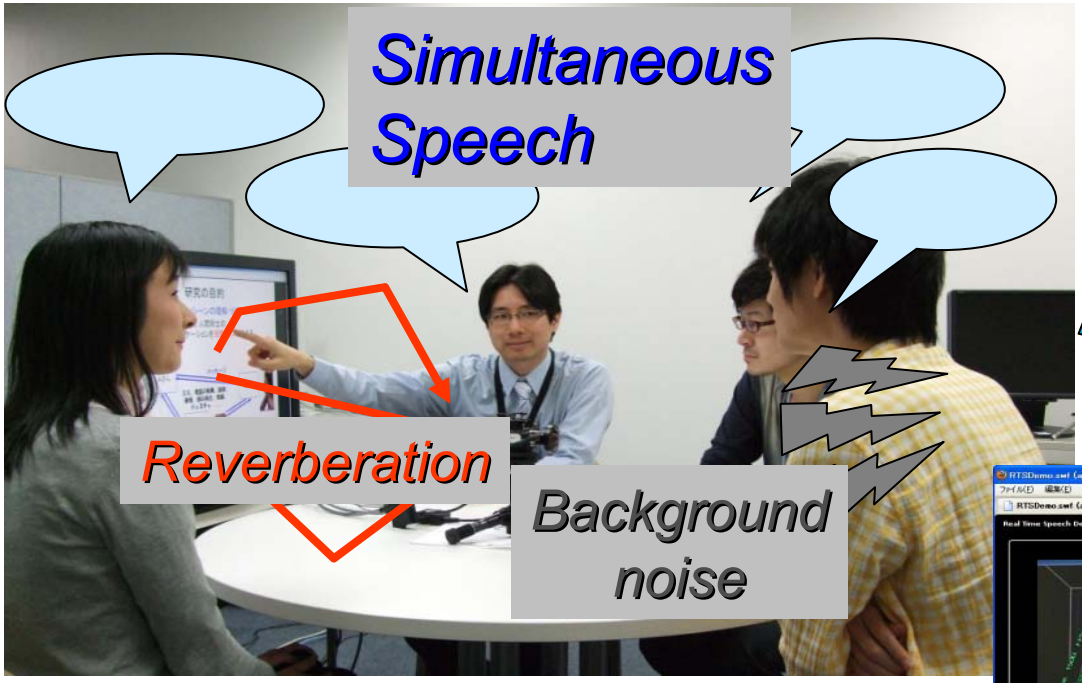
Step2:
Audio post-production
(de-noising, ***de-reverb***, sound effects)

Dereverberation system for audio post production [Kinoshita 2008]

- Dereverberation plug-in for Pro Tools: NML RevCon-RR (sold by TAC System, Inc.)



Online meeting recognition [Hori 2012]



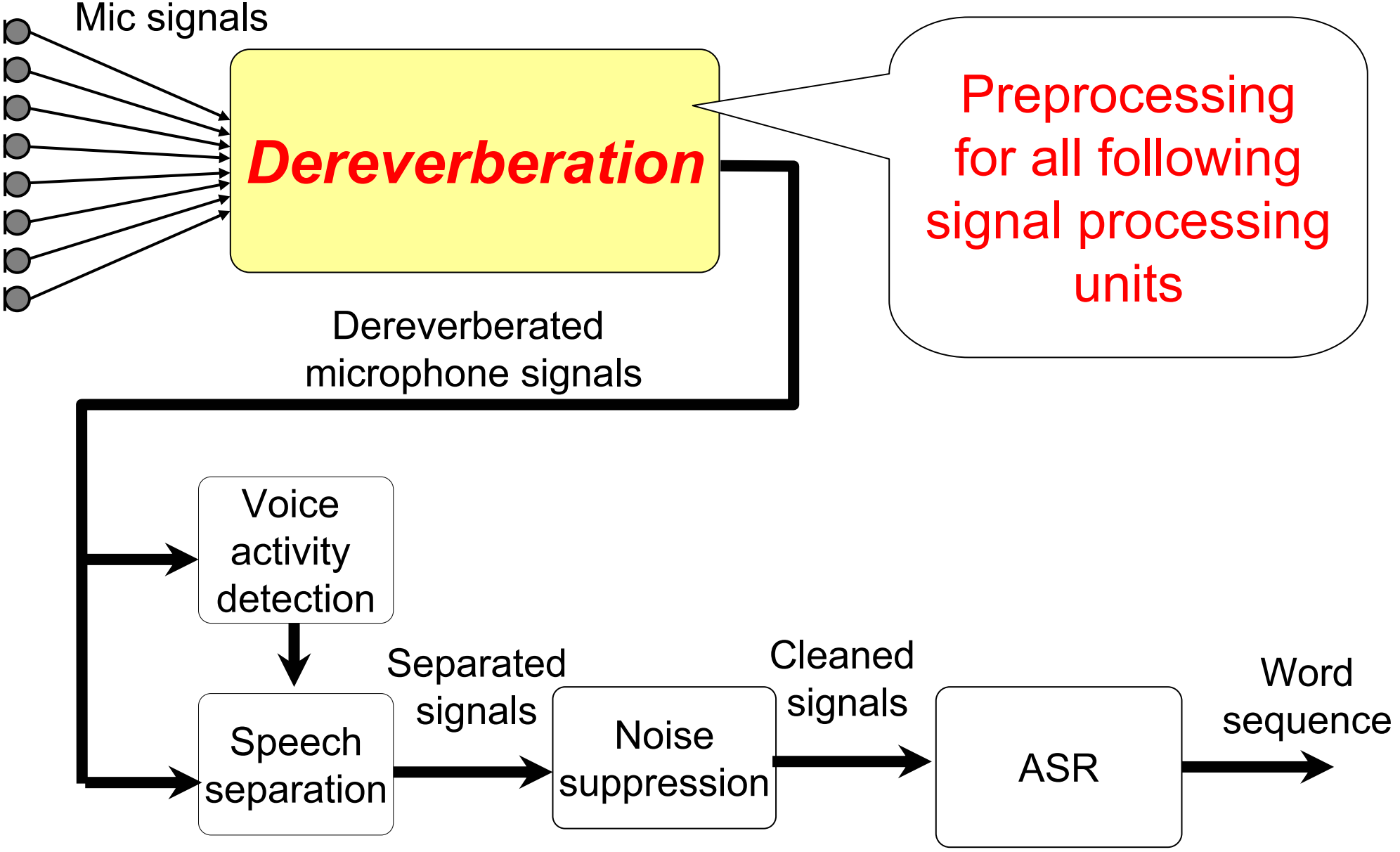
Recognize speech and other audio events

Real-time Meeting Browser

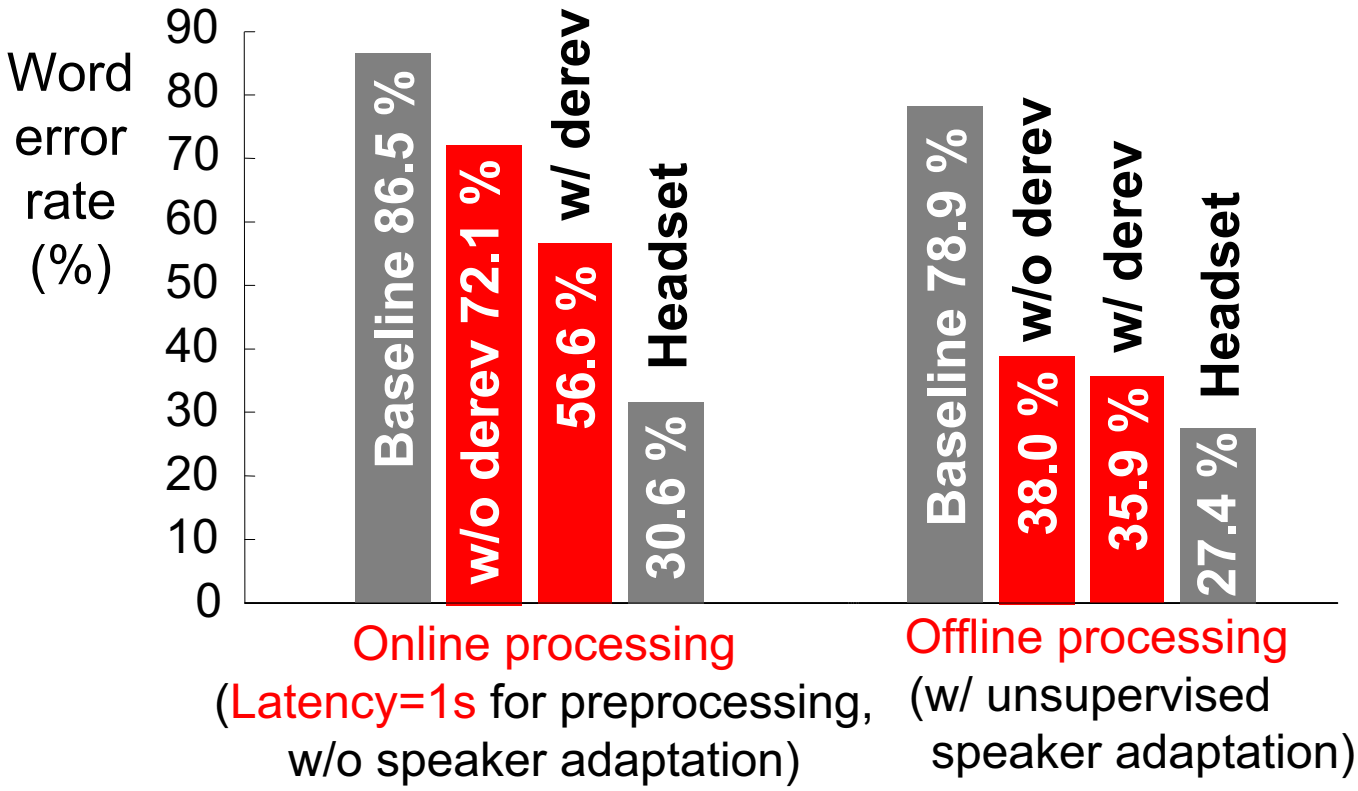
Who Spoke When, What, To-whom and How?

Show&Tell:
ST-3.2: Thursday, March 29, 10:30-12:30

Online/offline processing flow of meeting recognition



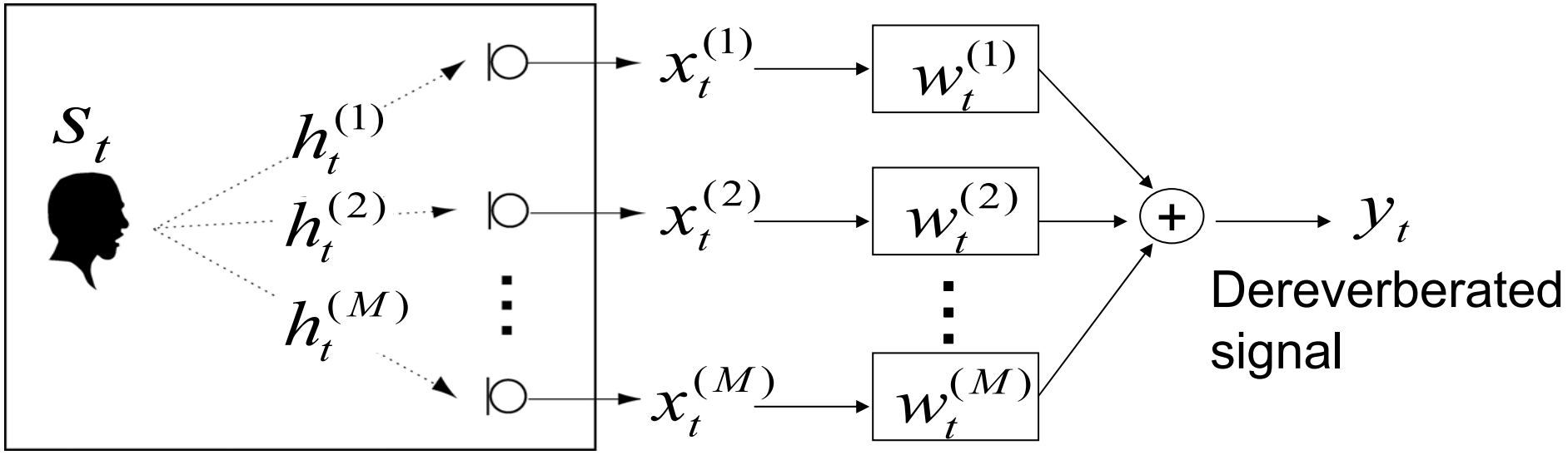
ASR performance w/ and w/o dereverberation



Baseline:
Distant microphone (w/o enhancement)
w/o derev:
BSS+denoise
w/ derev:
derev.+BSS+denoise
Headset:
Close microphone (w/o enhancement)

Test data:
Meeting by 4 speakers (15 min x 8 sessions)
Recording: 8 mics. (T_{60}: about 350 ms, Speaker-mic distance: 100 cm)
Acoustic model:
Trained on CSJ (corpus of spontaneous Japanese): headset recording
Language model:
Vocabulary size: 156K (LVCSR)

Questions to be answered



- What is inverse filtering ?
- Is the inverse filter robust against interferences ?
- Can we estimate the inverse filter with blind processing ?

Answers at a glance

What is inverse filtering ?

→ Inversion of room impulse responses (RIRs)

Is the inverse filter robust against interferences ?

→ Unfortunately no,
but there is a robust ‘approximate’ inverse filter

Can we estimate the inverse filter with blind processing ?

→ Yes, we can,
by using cues for distinguishing speech from RIRs

Part II. Multichannel blind inverse filtering

- Example applications
  - Professional audio post product
  - Meeting recognition with microp

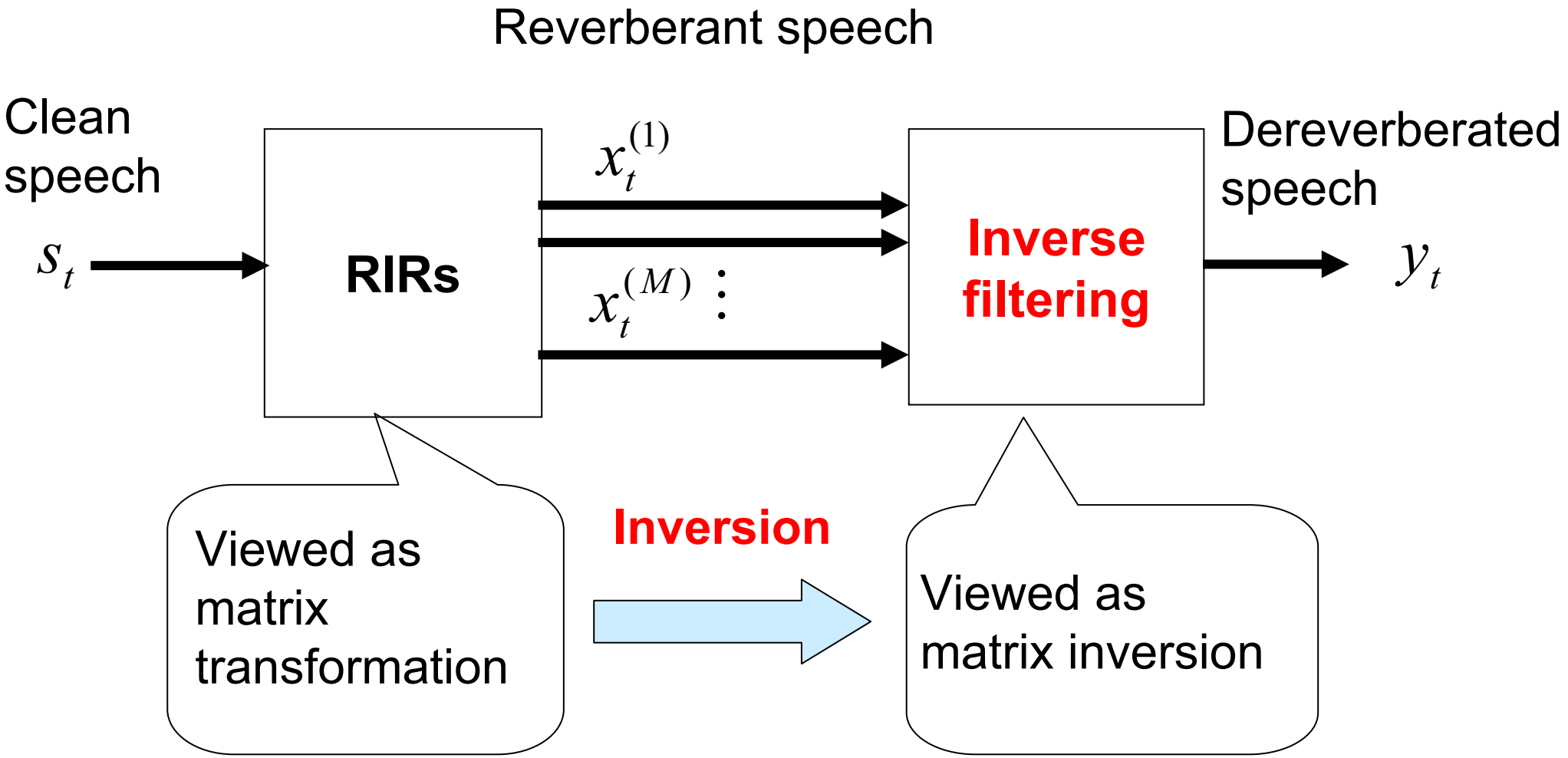
Assume ***non-blind processing*** for analysis purpose

- **Fundamentals: dereverberation with inverse filtering**
  - **What is inverse filter**
  - Robust 'approximate' inverse filter

- Blind inverse filtering
  - Overview of basic approaches
  - Closer look: multichannel linear prediction with time-varying source model

- Integration with blind source separation

Inversion of RIRs = Inversion of matrix transformation



Matrix/vector representations of RIR convolution/filtering

Single channel RIR convolution

$$\begin{bmatrix} x_t^{(m)} \\ x_{t-1}^{(m)} \\ \vdots \\ x_{t-K}^{(m)} \end{bmatrix} = \begin{bmatrix} h_0^{(m)} & h_1^{(m)} & \cdots & h_{L_h}^{(m)} & 0 & \cdots & 0 \\ 0 & h_0^{(m)} & h_1^{(m)} & & h_{L_h}^{(m)} & \ddots & \vdots \\ \vdots & \ddots & \ddots & \ddots & & \ddots & 0 \\ 0 & \cdots & 0 & h_0^{(m)} & h_1^{(m)} & \cdots & h_{L_h}^{(m)} \end{bmatrix} \begin{bmatrix} s_t \\ s_{t-1} \\ \vdots \\ s_{t-K_0} \end{bmatrix}$$

$$\mathbf{x}_t^{(m)} \quad \mathbf{H}^{(m)} \quad \mathbf{s}_t$$

$$\mathbf{x}_t^{(m)} = \mathbf{H}^{(m)} \mathbf{s}_t \qquad K = K_0 - L_h$$

Multichannel RIR convolution

$$\begin{bmatrix} \mathbf{x}_t^{(1)} \\ \vdots \\ \mathbf{x}_t^{(M)} \end{bmatrix} = \begin{bmatrix} \mathbf{H}^{(1)} \\ \vdots \\ \mathbf{H}^{(M)} \end{bmatrix} \mathbf{s}_t$$

$$\mathbf{x}_t \qquad \mathbf{H}$$

$$\mathbf{x}_t = \mathbf{H}\mathbf{s}_t$$

Single channel filtering

$$\sum_{\tau=0}^{K} w_\tau^{(m)} x_{t-\tau}^{(m)} = \underbrace{\left[w_0^{(m)}, \cdots, w_K^{(m)}\right]}_{\mathbf{w}^{(m)T}} \underbrace{\begin{bmatrix} x_t^{(m)} \\ \vdots \\ x_{t-K}^{(m)} \end{bmatrix}}_{\mathbf{x}_t^{(m)}}$$

$$= \mathbf{w}^{(m)T} \mathbf{x}_t^{(m)}$$

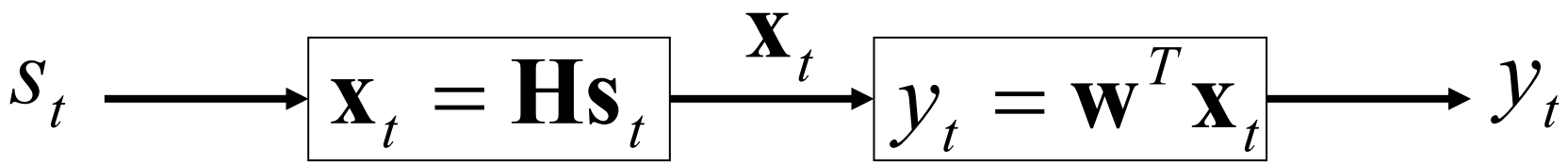
Multichannel filtering

$$\sum_{m=1}^{M} \mathbf{w}^{(m)T} \mathbf{x}_t^{(m)} = \underbrace{\left[\mathbf{w}^{(1)T} \quad \cdots \quad \mathbf{w}^{(M)T}\right]}_{\mathbf{w}^T} \begin{bmatrix} \mathbf{x}_t^{(1)} \\ \vdots \\ \mathbf{x}_t^{(M)} \end{bmatrix}$$

$$\mathbf{x}_t$$

$$y_t = \mathbf{w}^T \mathbf{x}_t$$

Existence of inverse filter



- A column vector $\mathbf{w}$ is an inverse filter when it satisfies:

$$s_t = y_t = \mathbf{w}^T\mathbf{H}\mathbf{s}_t \text{ where } \mathbf{s}_t = [s_t, s_{t-1}, \cdots, s_{t-K_0}]^T$$

$$\mathbf{w}^T\mathbf{H} = \mathbf{e}^T \text{ where } \mathbf{e} = [1, 0, \cdots, 0]^T$$

- An inverse filter $\mathbf{w}$ exists, when $\mathbf{H}$ is invertible, i.e., it is full column rank, and $\mathbf{w}$ is obtained as

$$\mathbf{w}^T = \mathbf{e}^T\mathbf{H}^+ \quad \text{where} \quad \mathbf{H}^+ = (\mathbf{H}^T\mathbf{H})^{-1}\mathbf{H}^T$$

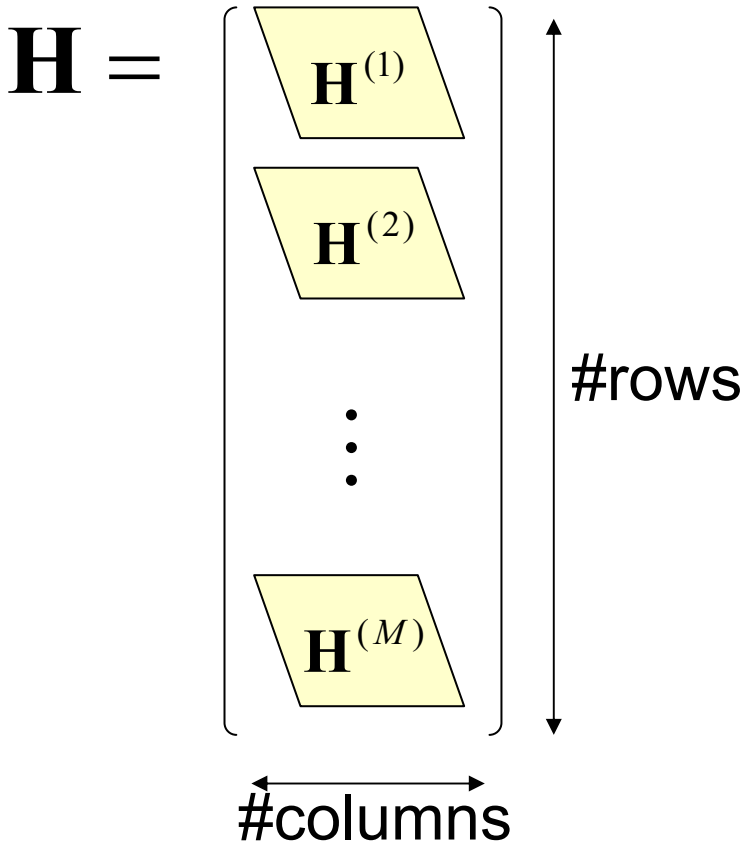
M (#mics) > 1 is required for single source case

- $\mathbf{H}$ is invertible, or full column rank, if and only if

  (#rows of $\mathbf{H}$) >= (#columns of $\mathbf{H}$)

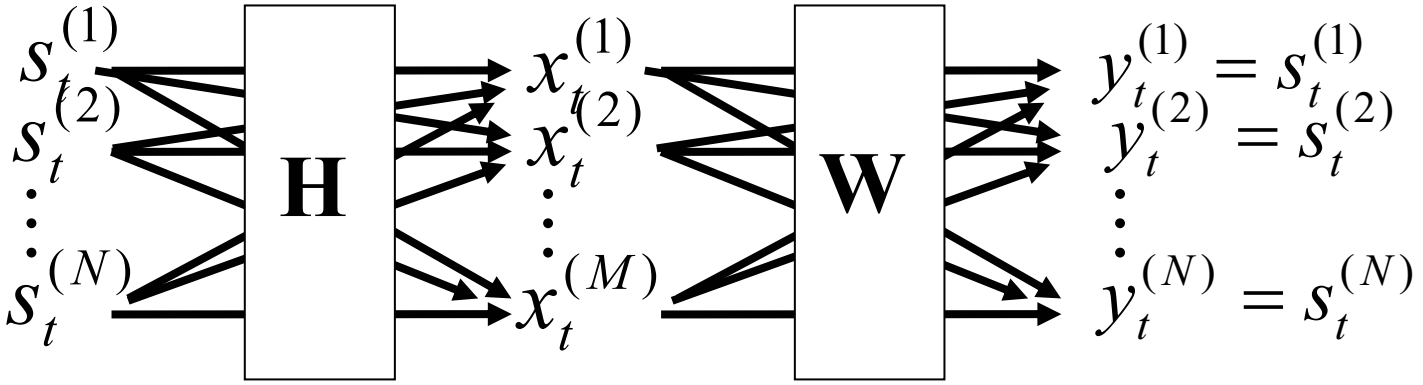
  and all columns are linearly independent
- In the case of single source, (#rows of $\mathbf{H}$) >= (#columns of $\mathbf{H}$) is satisfied if and only if

  $$M \text{ (\#mics)} > 1$$

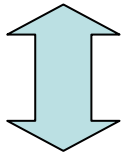


Generalization to N sources × M microphones case

Multiple-input/output inverse theorem (MINT) [Miyoshi 1988]



– Inverse filter exists when $\mathbf{H}$ is full column rank



Equivalent

- M (#mics) >N (#sources)
- $\mathbf{H}(z)$ does not contain common zeros

Part II. Multichannel blind inverse filtering

- Example applications
  - Professional audio post production
  - Meeting recognition with microphone arrays

- **Fundamentals: dereverberation with inverse filtering**
  - What is inverse filter
  - **Robust 'approximate' inverse filter**

- Blind inverse filtering
  - Overview of basic approaches
  - Closer look: multichannel linear prediction with time-varying source model

- Integration with blind source separation

Problem of inverse filtering

Assumptions for inverse filtering

- Invertible RIRs
- No additive noise
- Time-invariant RIRs

} Not realistic !

Inverse filter is too sensitive to modeling errors (noise or RIR change)

Inverse filter greatly amplifies noise

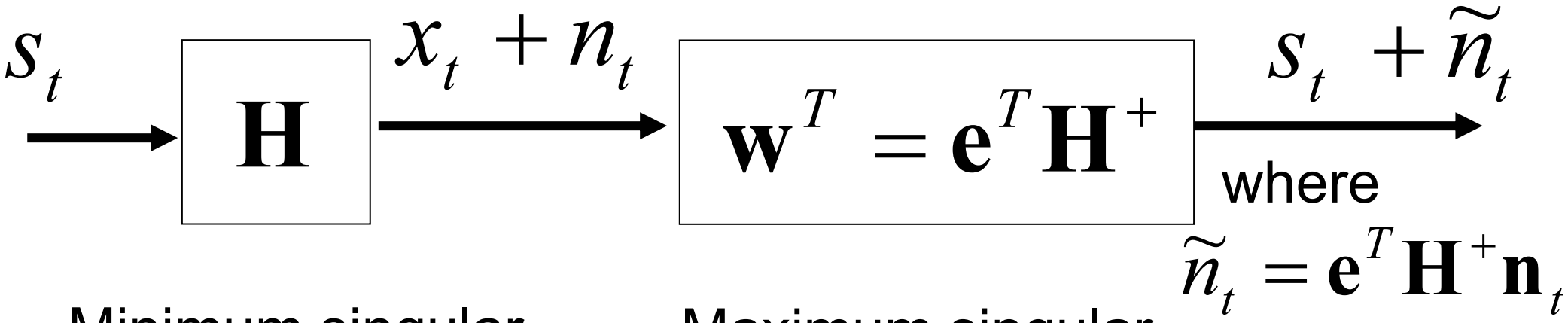
Noise-free reverberant case

- Clean speech
- Reverberant speech
  - Synthesized using a fixed RIR (RT60=0.5 s)
- Dereverberated speech using an inverse filter for known RIRs (2-channel)

Noisy reverberant case

- Noisy reverberant speech (SNR=30dB)
- Speech processed using the same inverse filter (2-channel)

Why inverse filter is so sensitive to additive noise



Minimum singular value of $\mathbf{H}$

Maximum singular value of $\mathbf{H}^+$

$\lambda_{\min}$

often extremely small

(compared to maximum singular value)

$\lambda_{\max}^{\text{inv}}$

often extremely large

Extremely amplifies noise

Standard numerical approach for robustness [Engl 1996]

- Regularization Noisy rev. Processed
  - A general technique for robust matrix inversion
    - Add a very small positive constant δ to diagonal of $\mathbf{H}^T\mathbf{H}$ for calculating the pseudo-inversion of $\mathbf{H}$

$$\mathbf{H}^+ = (\mathbf{H}^T\mathbf{H})^{-1}\mathbf{H}^T \Rightarrow \tilde{\mathbf{H}}^+ = (\mathbf{H}^T\mathbf{H} + \delta\,\mathbf{I})^{-1}\mathbf{H}^T$$

$\mathbf{I}$: Identity matrix

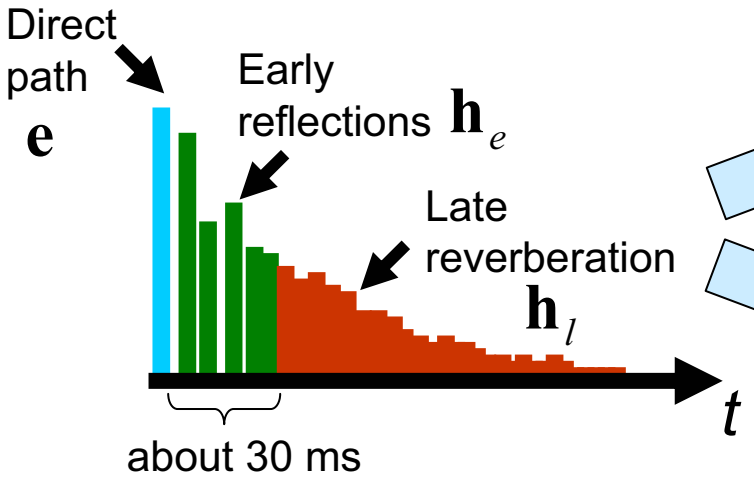
  - It can reduce the maximum singular value $\lambda_{\max}^{\text{inv}}$ of $\tilde{\mathbf{H}}^+$

⇒ Noise amplification is greatly mitigated

Room acoustics motivated approach for robustness

- Channel shortening Noisy rev. Processed
  - Set “direct signal + early reflections” as target signal, and reduce only late reverberation

Illustration of an RIR

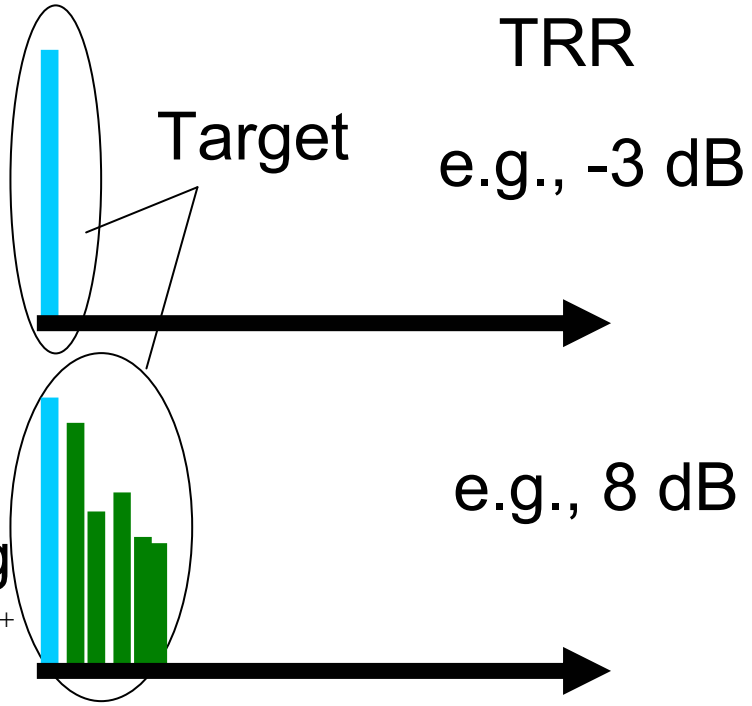


Inverse filtering

$$\mathbf{w}^T = \mathbf{e}^T \tilde{\mathbf{H}}^+$$

Channel shortening

$$\mathbf{w}^T = (\mathbf{e}^T + \mathbf{h}_e^T)\tilde{\mathbf{H}}^+$$



Target to reverberation ratio (TRR) w/ channel shortening is much higher than TRR w/ inverse filtering

Intermediate summary II-1

- Dereverberation: inversion of RIRs
  - Assuming RIRs to be a time-invariant linear system
- Inverse filter exists
  - When we have more microphones than sources
  - But it may be very sensitive to additive noise
- 'Approximate' inverse filter is robust against noise
  - Based on regularization and channel shortening

Part II. Multichannel blind inverse filtering

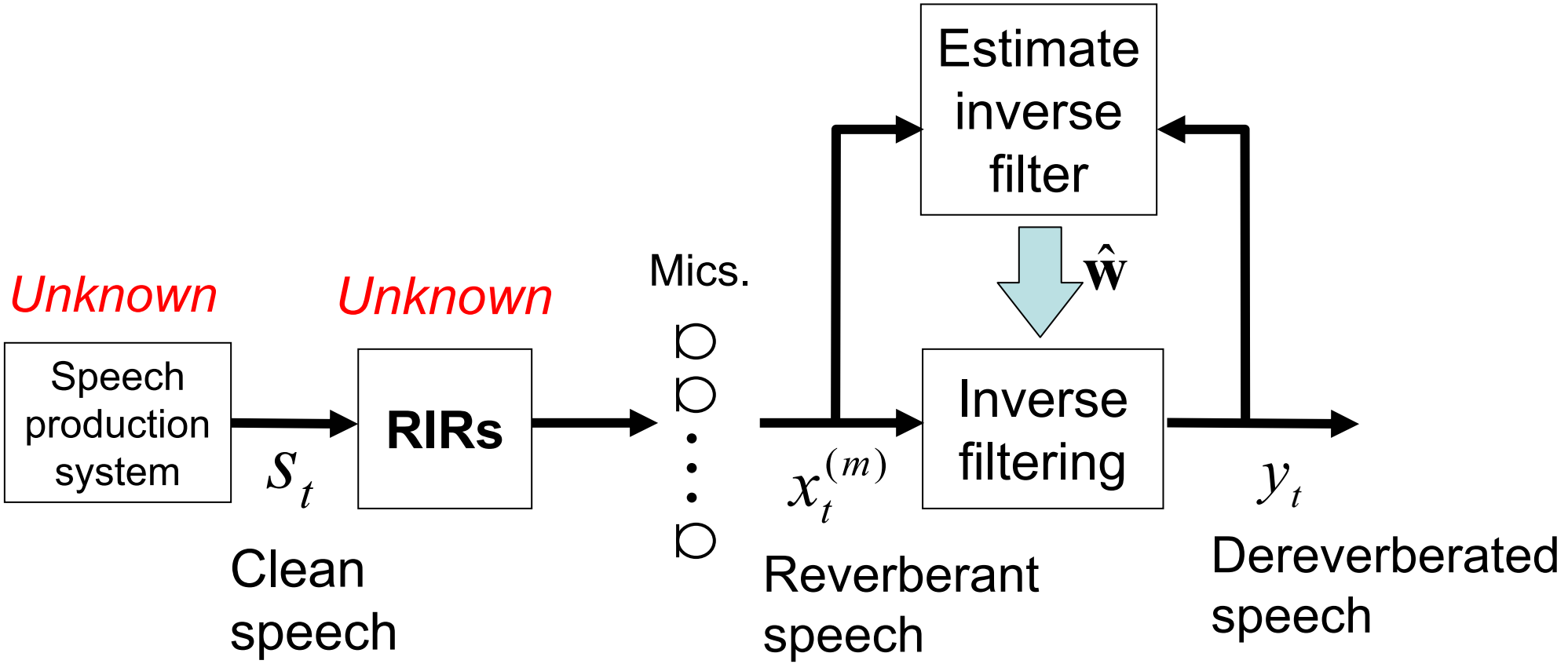
- Example applications
  - Professional audio post production
  - Meeting recognition with microphone arrays

- Fundamentals: dereverberation with inverse filtering
  - What is inverse filter
  - Robust ‘approximate’ inverse filter

- **Blind inverse filtering**
  - **Overview of basic approaches**
  - Closer look: multichannel linear prediction with time-varying source model

- Integration with blind source separation

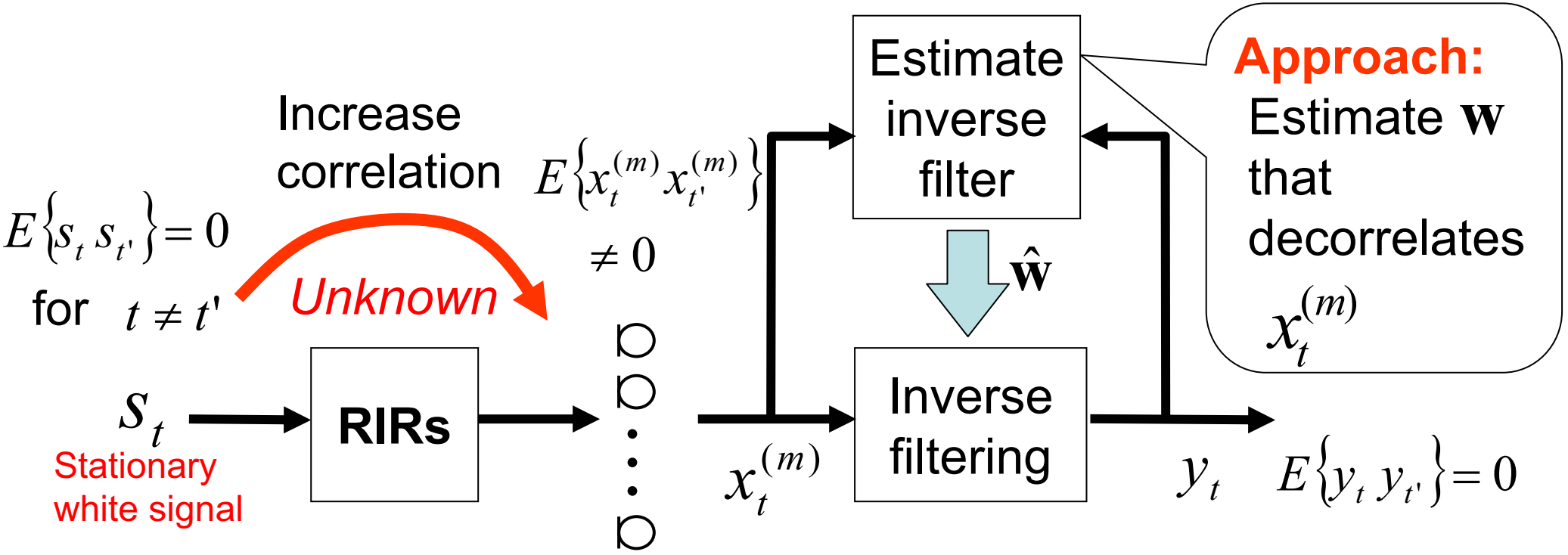
Blind inverse filtering based dereverberation



Two approaches

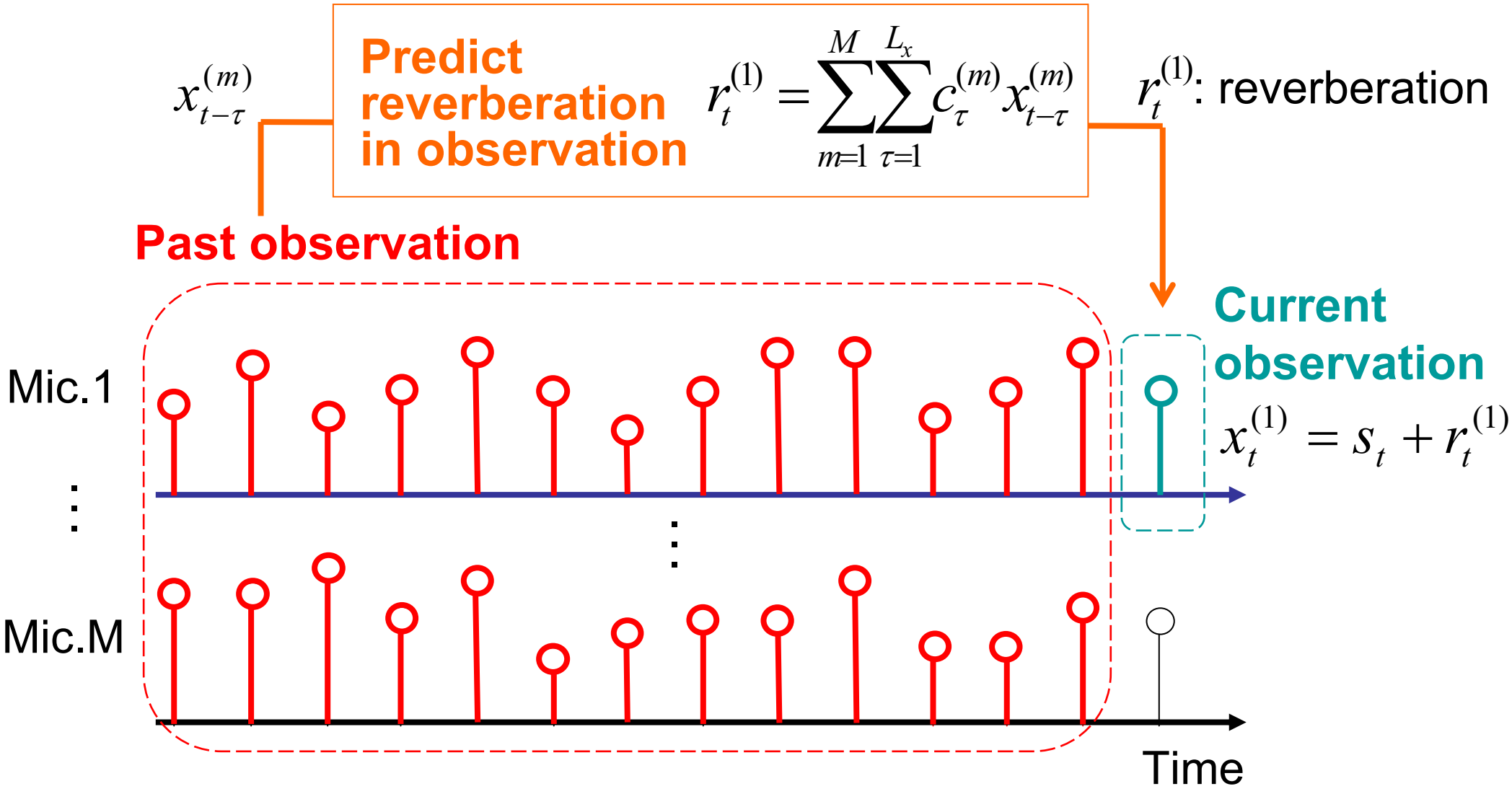
- RIR estimation + RIR inversion
- ***Direct estimation of inverse filter***

Conventional decorrelation approaches for stationary white signal



- SOS approach assumes s_t to be stationary white Gaussian
  - ***Multichannel linear prediction (MCLP)*** [Slock 1994], [Abed-Meraim 1997]
- HOS approach assumes s_t to be an i.i.d. sequence
  - Higher order decorrelation [Sato 1975], [Bellini 1994]

Multichannel linear prediction (MCLP)



MCLP based decorrelation [Slock 1994], [Abed-Meraim 1997]

- $x_t^{(1)}$ is modeled by

$$x_t^{(1)} = \sum_{m=1}^{M}\sum_{\tau=1}^{K} c_\tau^{(m)} x_{t-\tau}^{(m)} + s_t = \mathbf{x}_{t-1}^T \mathbf{c} + s_t \quad \Rightarrow \quad s_t = x_t^{(1)} - \mathbf{x}_{t-1}^T \mathbf{c}$$

Predicted signal (= reverberation)

Prediction error (= direct signal)

where $\mathbf{c} = [c_1^{(1)}, \cdots, c_K^{(1)}, \cdots\cdots, c_1^{(M)}, \cdots, c_K^{(M)}]^T$ is prediction coeffs.

  - $\mathbf{c}$ ***is equivalent to*** inverse filter $\mathbf{w}$

- $\mathbf{c}$ can be estimated by minimizing prediction error when sources are stationary and uncorrelated in time

$$\hat{\mathbf{c}} = \underset{\mathbf{c}}{\operatorname{argmin}} \sum_t \left(x_t^{(1)} - \mathbf{x}_{t-1}^T \mathbf{c} \right)^2$$

  - Quadratic form: ***optimized using a closed form solution***

Why dereverberation can be achieved by MCLP

$$\sum_{t=1}\left(x_t^{(1)}-\mathbf{c}^T\mathbf{x}_{t-1}\right)^2=\sum_{t=1}\left(\sum_{\tau=0}h_\tau^{(1)}s_{t-\tau}-\mathbf{x}_{t-1}^T\mathbf{c}\right)^2$$

$$=\sum_{t=1}\left(s_t+\sum_{\tau=1}h_\tau^{(1)}s_{t-\tau}-\mathbf{x}_{t-1}^T\mathbf{c}\right)^2$$

$$=\sum_{t=1}|s_t|^2+\sum_{t=1}\left(\sum_{\tau=1}h_\tau^{(1)}s_{t-\tau}-\mathbf{x}_{t-1}^T\mathbf{c}\right)^2$$

$$\geq\sum_{t=1}|s_t|^2$$

$h_0^{(1)}=1$ is usually assumed for MCLP without loss of generality

$\sum_{t=1}s_ts_{t'}=0 \quad for \quad t\neq t'$

(and thus $\sum_{t=1}s_t\mathbf{x}_{t-1}^T=0$)

Minimization is achieved only when

True reverberation $\sum_{\tau=1}h_\tau^{(1)}s_{t-\tau}$ **=** **Predicted reverberation** $\mathbf{x}_{t-1}^T\mathbf{c}$

Robustness of MCLP against noise

Let $z_t^{(m)} = x_t^{(m)} + n_t^{(m)}$ be noisy reverberant observation.

Additive noise (or can be viewed as modeling error)

Assume $x_t^{(m)}$ and $n_t^{(m)}$ to be uncorrelated,
then, the cost function becomes

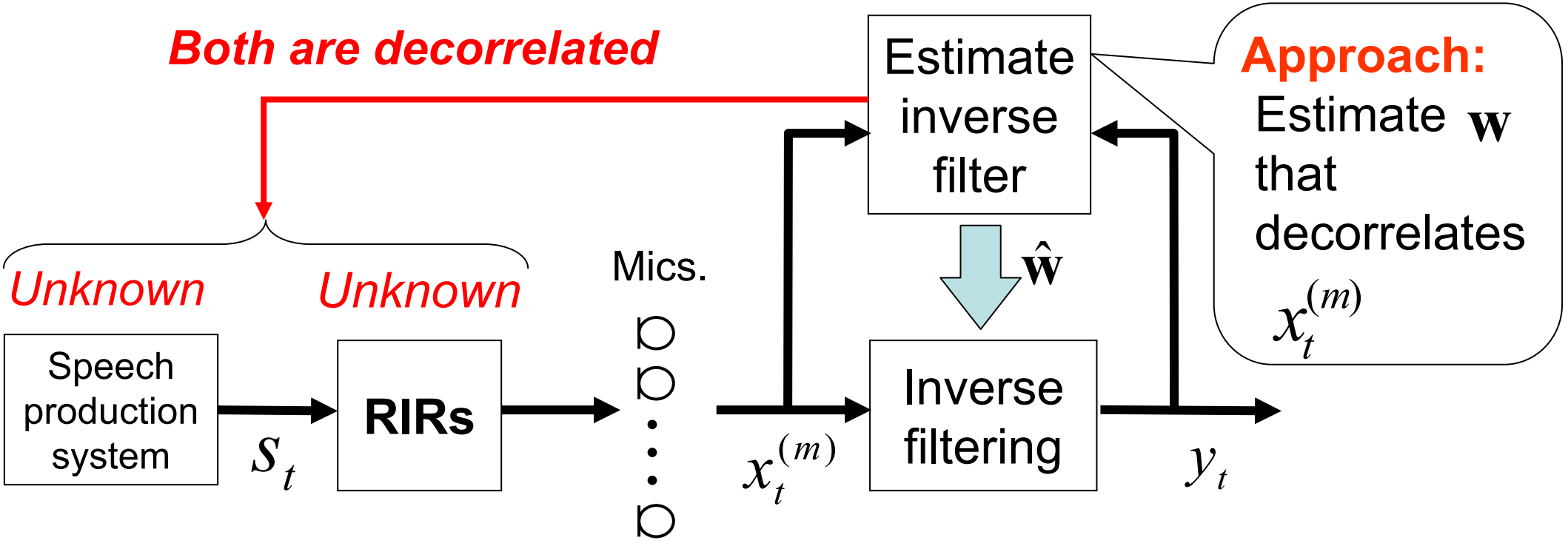
$$\sum_{t=1}\left(z_t^{(1)} - \mathbf{z}_{t-1}^T\mathbf{c}\right)^2 = \sum_{t=1}\left(x_t^{(1)} - \mathbf{x}_{t-1}^T\mathbf{c}\right)^2 + \sum_{t=1}\left(n_t^{(1)} - \mathbf{n}_{t-1}^T\mathbf{c}\right)^2$$

Cost function for dereverberation

Cost function for noise amplification

Regularization is inherently included

Problem of decorrelation approach for speech dereverberation



Problem:
Not only dereverberate $x_t^{(m)}$ but also decorrelate s_t

Key to the solution:
Use cues to separate speech and RIRs

Cues for separating speech and RIRs

Cues	Speech	RIRs
Inter-channel difference	**Common** to all the microphone signals	**Different** for each microphone
Auto-correlation duration	Correlated only within short time interval of the order of **30 ms**	Correlated within long time interval over **100 ms**
Nonstationarity	Stationary only within short time period of the order of **30 ms**	Stationary over long time period of the order of **1000 ms or larger**

Approaches to blind inverse filtering

Cues

- ***Subspace method*** (RIR estimation + inversion)
  - [Furuya 1997], [Gannot 2003], [Gaubitch 2006]

  } Inter-channel difference

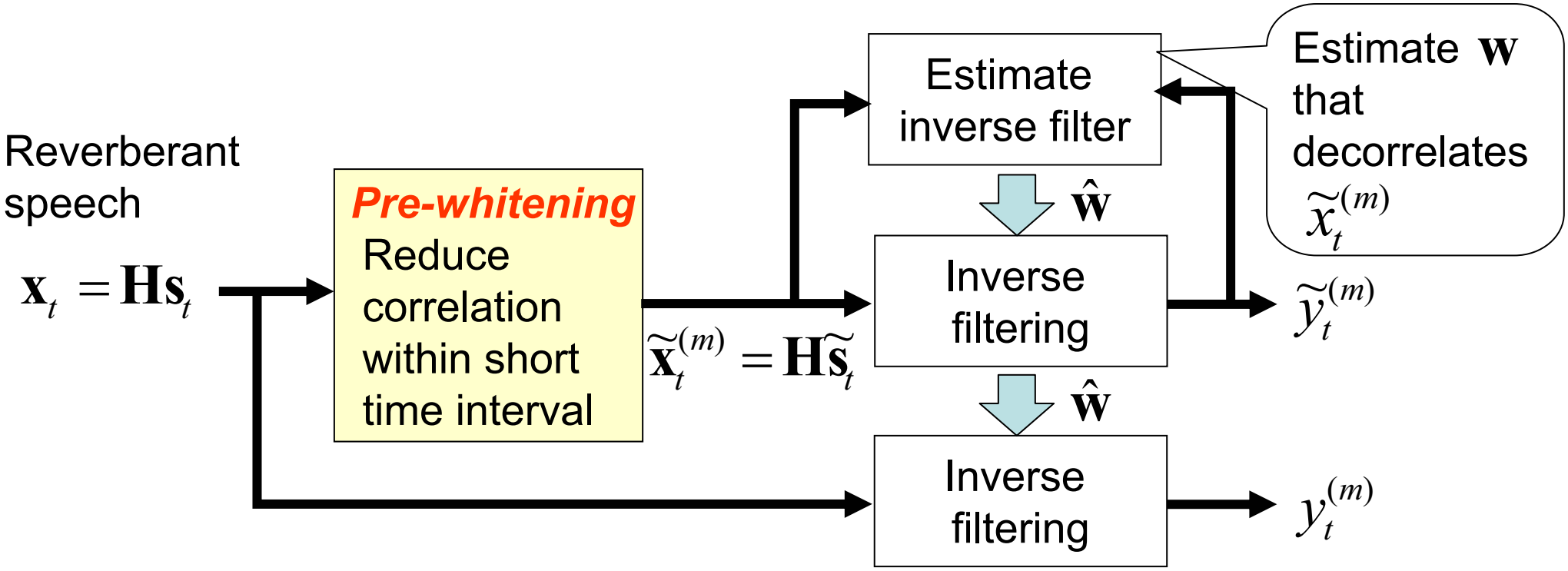
- ***Pre-whitening + decorrelation***
  - Second-order statistics (SOS): [Gaubitch 2003], [Furuya 2007], [Triki 2007]
  - Higher-order statistics (HOS): [Gillespie 2001]
- ***Channel shortening***
  - [Gillespie 2003], [Kinoshita 2009]

  } Auto-correlation duration

- ***Joint speech and reverberation modeling***
  - [Hopgood 2003], [Buchner (TRINICON) 2010], [Yoshioka 2007], [Nakatani 2008]

  } Auto-correlation duration and nonstationarity

Pre-whitening + decorrelation

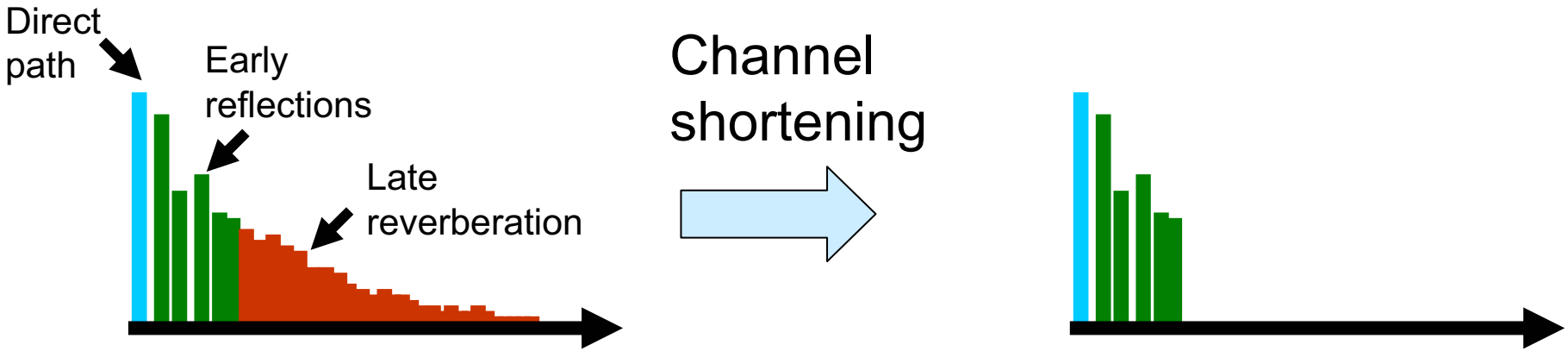


- A typical method for pre-whitening
  - Low-dimensional (e.g., 12-dim) single channel linear prediction often used

Assumption: pre-whitening can decorrelate only $\mathbf{s}_t$ in $\mathbf{x}_t = \mathbf{H}\mathbf{s}_t$, and we can obtain $\tilde{\mathbf{x}}_t = \mathbf{H}\tilde{\mathbf{s}}_t$ where $\tilde{\mathbf{s}}_t$ is an unknown decorrelated speech

Channel shortening

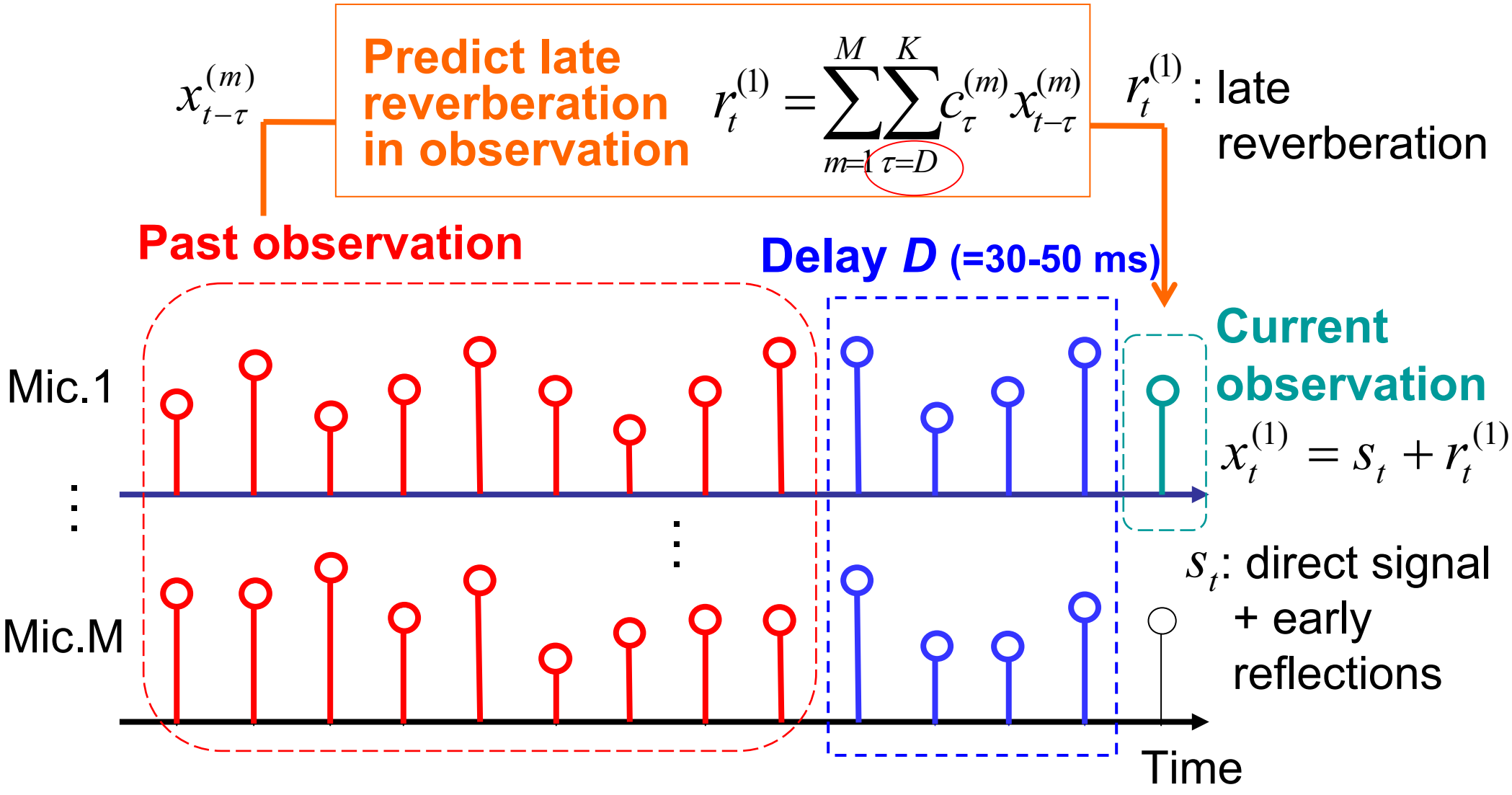
- Introduce constraints so that dereverberation reduces only late reverberation



Make derev. ***robust*** and ***do not decorrelate speech***

- Techniques:
  - Correlation shaping [Gillespie 2003]
  - Multistep MCLP [Kinoshita 2009]

Multistep MCLP [Gesbert 1997], [Kinoshita 2009]



Approaches to blind inverse filtering

Cues

- ***Subspace method*** (RIR estimation + inversion)
  - [Furuya 1997], [Gannot 2001], [Gaubitch 2006]

  } Spatial diversity

- ***Pre-whitening + decorrelation***
  - Second-order statistics (SOS): [Gaubitch 2003], [Furuya 2006], [Triki 2006]
  - Higher-order statistics (HOS): [Gillespie 2001]
- ***Channel shortening***
  - [Gillespie 2003], [Kinoshita 2006]

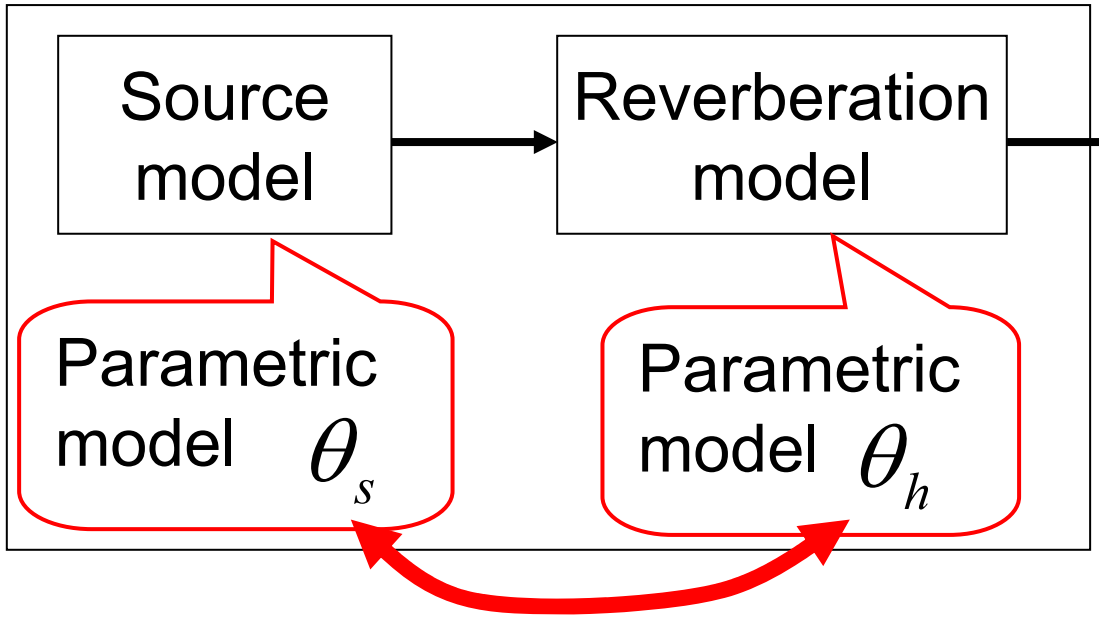
  } Duration of auto-correlation

- ***Joint speech and reverberation modeling***
  - [Hopgood 2003], [Buchner (TRINICON) 2010], [Yoshioka 2007], [Nakatani 2008]

  } Duration of auto-correlation and nonstationarity

Joint speech and reverberation modeling for derev.

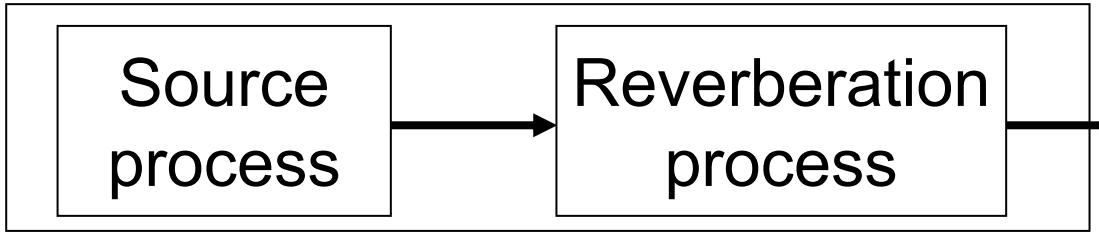
Model of generative system



$$x_t \sim p(x_t;\theta) \quad \theta = \{\theta_s, \theta_h\}$$

Parameter estimation by

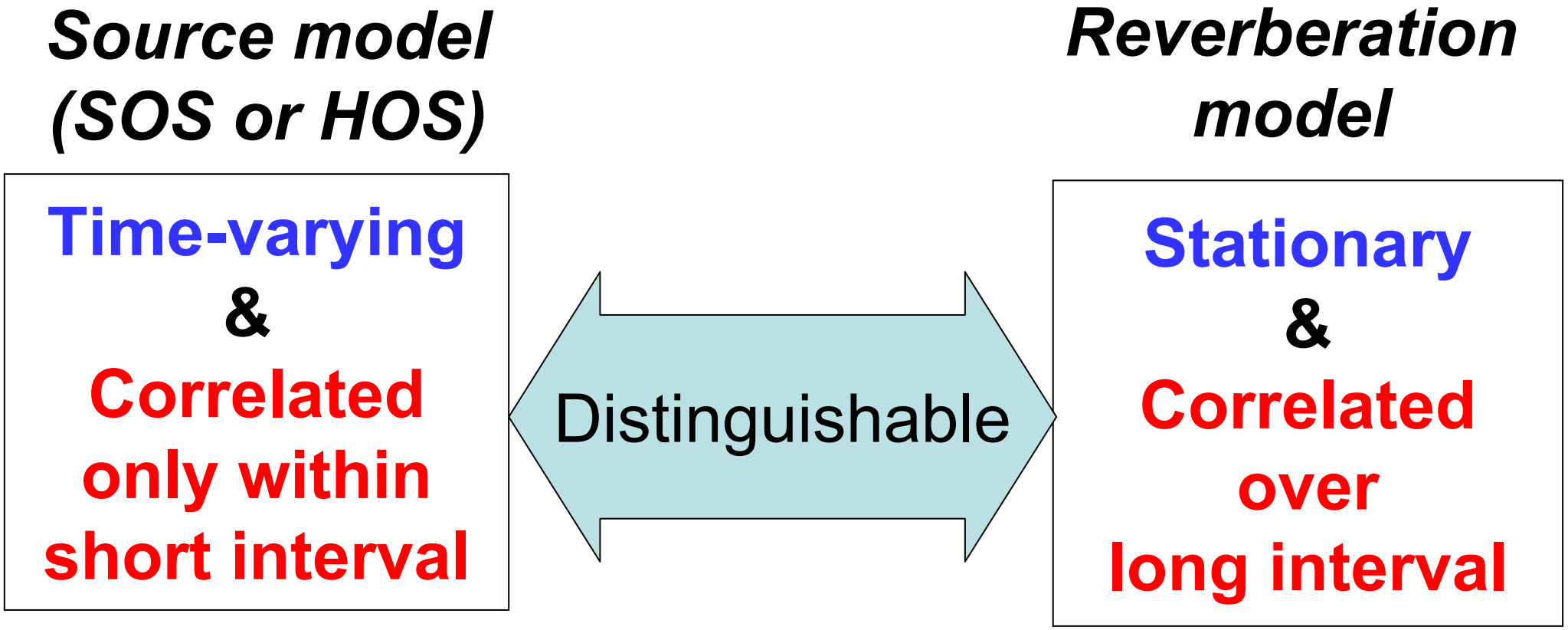
- **Likelihood maximization**
  [Hopgood 2003], [Yoshioka 2007], [Nakatani 2008]
- **Kullback-Leibler divergence minimization**
  [Buchner (TRINICON) 2010]



x_t Reverberant observation

Unknown true generative system

Models for source process and reverberation process



Multichannel blind partial deconvolution (MCBPD) by TRINICON

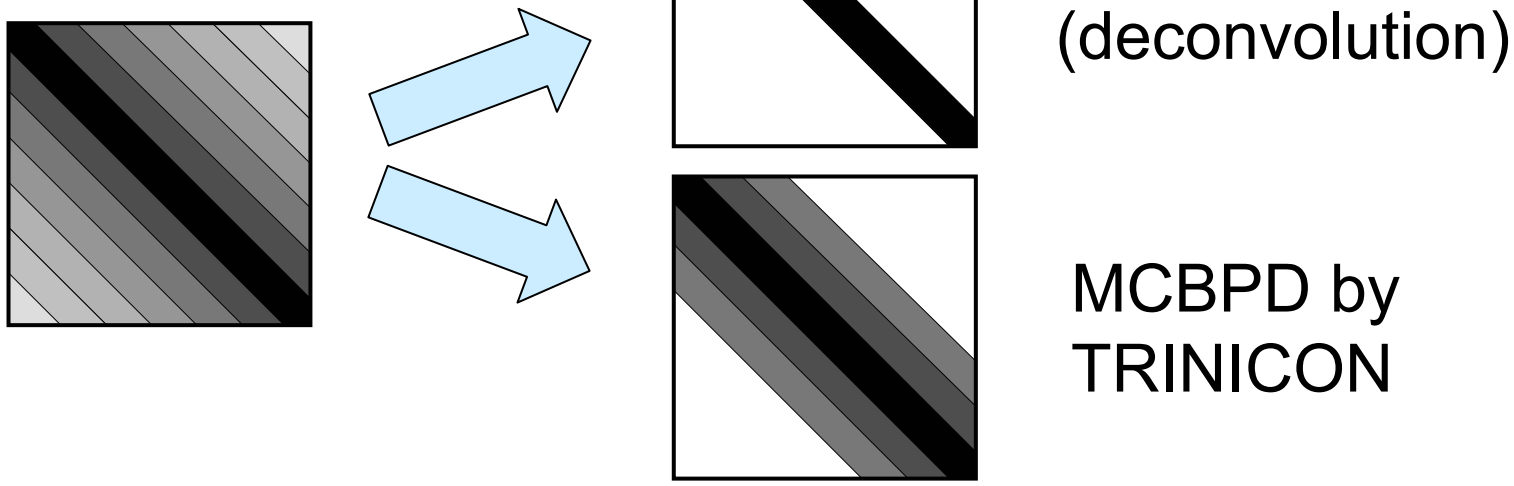
Cost function for SOS-TRINICON [Buchner 2010]

$$\mathcal{J}_{\text{SOS}} = \sum_t \left\{ \log\det \hat{\mathbf{R}}_{s,t} - \log\det \hat{\mathbf{R}}_{y,t} \right\}$$

Goal — Autocorrelation matrix of y_t

Autocorrelation matrix of observed signal

Goal $\hat{\mathbf{R}}_{s,t}$



Decorrelation (deconvolution)

MCBPD by TRINICON

Part II. Multichannel blind inverse filtering

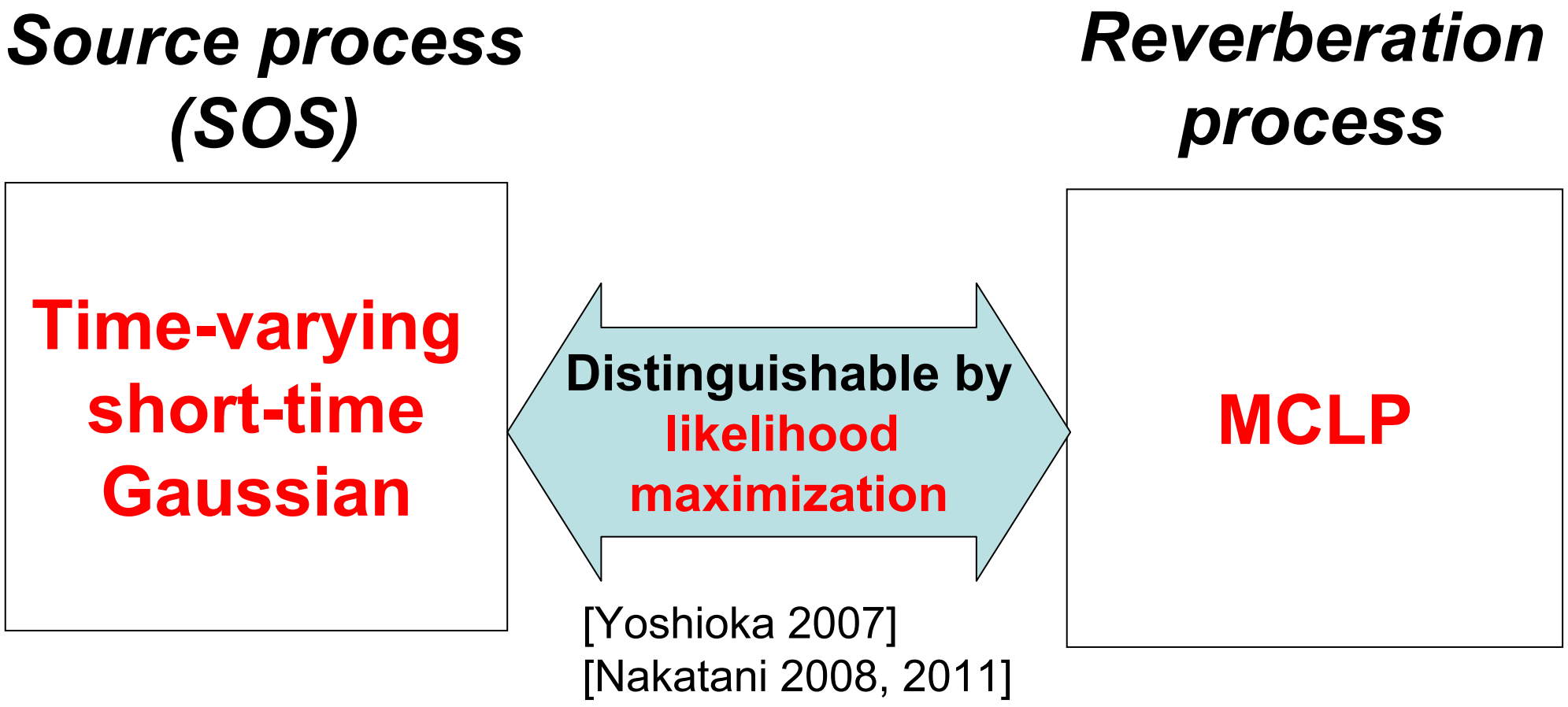
- Example applications
  - Professional audio post production
  - Meeting recognition with microphone arrays

- Fundamentals: dereverberation with inverse filtering
  - What is inverse filter
  - Robust ‘approximate’ inverse filter

- **Blind inverse filtering**
  - Overview of basic approaches
  - **Closer look: multichannel linear prediction with time-varying source model**

- Integration with blind source separation

MCLP with time-varying source model for dereverberation



Reformulation of MCLP based on likelihood maximization

$$L(\mathbf{c}) = \log p_x(\{\mathbf{x}_t\};\mathbf{c})$$

$$= \sum_{t=1} \log p_x(x_t^{(1)} \mid \{x_{t'}\}_{t'=1:t-1};\theta) + const.$$

$$= \sum_{t=1} \log p_s(s_t) + const. \quad \text{where } s_t = x_t^{(1)} - \mathbf{x}_{t-1}^T \mathbf{c}$$

Source model

Conditional probability rule

$$x_t^{(1)} = \mathbf{x}_{t-1}^T \mathbf{c} + s_t$$

Assume $p_s(s_t) = N(s_t;0,1)$ (**stationary white Gaussian**), then

$$L(\mathbf{c}) = -(1/2)\sum_{t=1} | x_t^{(1)} - \mathbf{c}^T \mathbf{x}_{t-1} |^2 + const.$$

$$\max_{\mathbf{c}} L(\mathbf{c}) \quad \Leftrightarrow \quad \min_{\mathbf{c}} \sum_{t=1} | x_t^{(1)} - \mathbf{c}^T \mathbf{x}_{t-1} |^2$$

Maximize likelihood — **Minimize prediction error**

Time-varying Gaussian source model (TVGSM)

1. Each short time segment $\bar{\mathbf{s}}_t = \overbrace{[s_t\ s_{t-1} \cdots s_{t-N+1}]}^{30\ \text{ms order}}{}^T$ is ***stationary multivariate Gaussian***, which can be characterized by

$$p_s(\bar{\mathbf{s}}_t; \mathbf{R}_t) = \mathcal{N}(\bar{\mathbf{s}}_t; 0, \mathbf{R}_t)$$

where $\mathbf{R}_t = E\{\bar{\mathbf{s}}_t \bar{\mathbf{s}}_t^T\}$ is an autocorrelation matrix

2. $\mathbf{R}_t$ varies over different time segments

$$\theta_s = \{\mathbf{R}_t\} : \text{parameters to be estimated}$$

MCLP with multivariate source model

$$x_t^{(1)} = \mathbf{x}_{t-1}^T \mathbf{c} + s_t$$

$$\Rightarrow \quad \bar{\mathbf{x}}_t^{(1)} = \mathbf{X}_{t-1}\mathbf{c} + \bar{\mathbf{s}}_t \quad \text{or}$$

$$\underbrace{\begin{bmatrix} x_t^{(1)} \\ x_{t-1}^{(1)} \\ \vdots \\ x_{t-N+1}^{(1)} \end{bmatrix}}_{\bar{\mathbf{x}}_t^{(1)}} = \underbrace{\begin{bmatrix} \mathbf{x}_{t-1}^T \\ \mathbf{x}_{t-2}^T \\ \vdots \\ \mathbf{x}_{t-N}^T \end{bmatrix}}_{\mathbf{X}_{t-1}} \mathbf{c} + \underbrace{\begin{bmatrix} s_t \\ s_{t-1} \\ \vdots \\ s_{t-N+1} \end{bmatrix}}_{\bar{\mathbf{s}}_t}$$

30 ms order

- Prediction error $\bar{\mathbf{s}}_t$ is assumed to follow TVGSM

Likelihood function of MCLP with TVGSM

$$L(\mathbf{c}, \{\mathbf{R}_t\}) = \sum_t \log p_s(\bar{\mathbf{s}}_t; \mathbf{c}, \{\mathbf{R}_t\})$$

where $\bar{\mathbf{s}}_t = \bar{\mathbf{x}}_t^{(1)} - \mathbf{X}_{t-1}\mathbf{c}$ and $p_s(\bar{\mathbf{s}}_t; \mathbf{R}_t) = \mathcal{N}(\bar{\mathbf{s}}_t; 0, \mathbf{R}_t)$

$$L(\mathbf{c}, \{\mathbf{R}_t\}) = -\sum_t \| \bar{\mathbf{x}}_t^{(1)} - \mathbf{X}_{t-1}\mathbf{c} \|_{\mathbf{R}_t} - \log |\mathbf{R}_t|$$

Prediction error weighted by $\mathbf{R}_t^{-1}$

Normalization term

where $\| \bar{\mathbf{s}} \|_{\mathbf{R}} = \bar{\mathbf{s}}^T \mathbf{R}^{-1} \bar{\mathbf{s}}$ (quadratic form)

Iterative optimization procedure

Initialize

$$\hat{\mathbf{R}}_t = E\{\bar{\mathbf{x}}_t^T \bar{\mathbf{x}}_t\}$$

Closed form

Update prediction coeffs. $\hat{\mathbf{c}}$

$$\hat{\mathbf{c}} = \arg\min_{\mathbf{c}} \sum_t \| \bar{\mathbf{x}}_t^{(1)} - \mathbf{X}_{t-1}\mathbf{c} \|_{\hat{\mathbf{R}}_t}$$

$\hat{\mathbf{c}}$ ↓ ↑ $\{\hat{\mathbf{R}}_t\}$

Update source model $\{\hat{\mathbf{R}}_t\}$

1. Dereverberate $\{\bar{\mathbf{x}}_t\}$

$$\hat{\bar{\mathbf{s}}}_t = \bar{\mathbf{x}}_t^{(1)} - X_{t-1}\hat{\mathbf{c}}$$

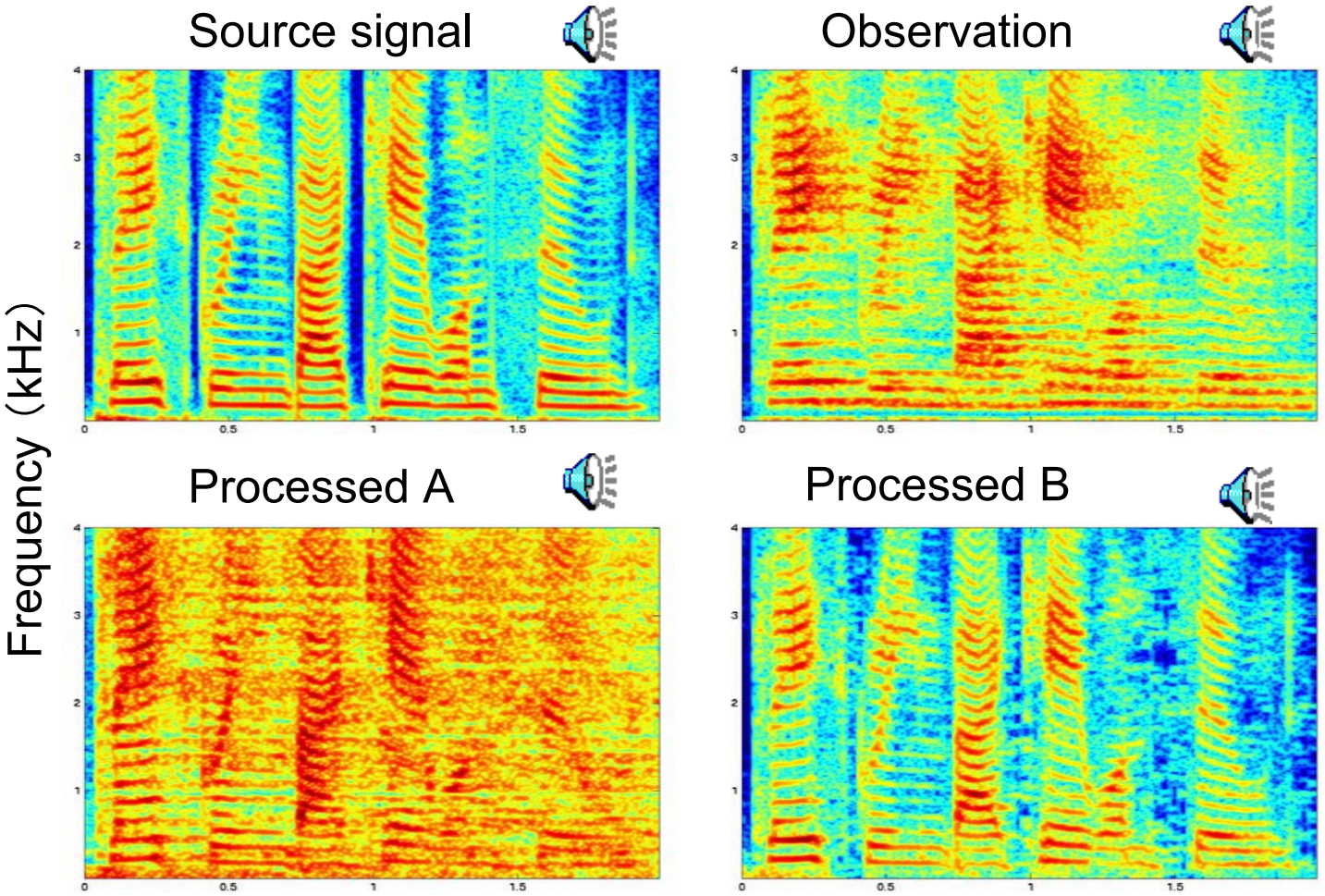
2. Calculate autocorrelation matrix of $\hat{\bar{\mathbf{s}}}_t$

$$\hat{\mathbf{R}}_t = E\{\hat{\bar{\mathbf{s}}}_t^T \hat{\bar{\mathbf{s}}}_t\}$$

$\{\hat{\bar{\mathbf{s}}}_t\}$ ↓

A few iterations are sufficient for convergence

Importance of time-varying source model



(A) MCLP with stationary white Gaussian source model

(B) MCLP with TVGSM

T_{60} : 0.5 s
Source-mic distance: 1.5 m
mics : 2

Recording: 2.5 s

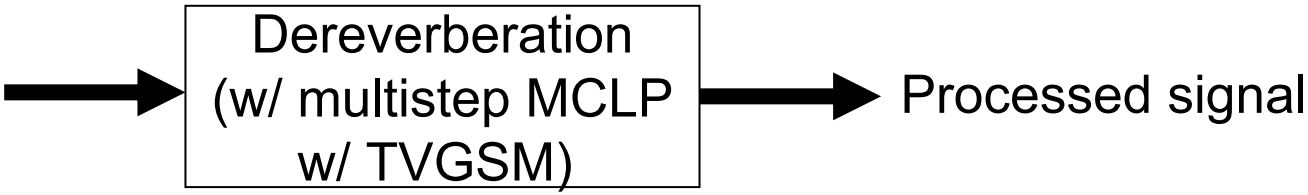
A few seconds of observation are sufficient for dereverberation

Blind inverse filtering works in noisy environments

Noisy reverberant speech

mics: 8

source-mics distance: 2 m



SNR	TRR*
15 dB	5.8 dB (T_{60} =0.39s)
10 dB	
15 dB	0.1 dB (T_{60} =0.65s)
10 dB	

Reduce reverberation

TRR
15.2 dB
13.8 dB
11.4 dB
10.3 dB

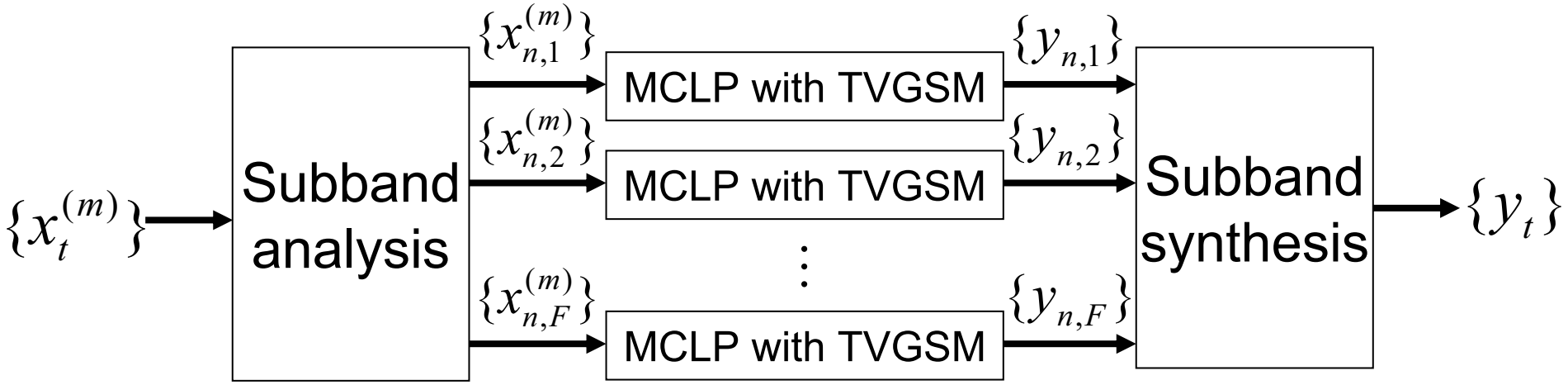
Noise: additive white noise (reproduced and recorded by 8 mics)

Noise may slightly increase, but not significantly

*TRR: Target-to-reverberation ratio (target = direct signal + early reflections)

Computationally efficient implementation

- Subband decomposition approach [Nakatani 2010], [Yoshioka 2009b]



- Computational efficiency largely improves

Real-time factor (RTF) using MATLAB (RT60: 0.5 s, # mics: 2)	Time-domain	Subband
	170	**0.8**

Processing flow with subband decomposition [Nakatani 2010]

1. Set analysis parameters

D : prediction delay (D should be # of subband samples corresponding to 30 ms, or larger)

L : length of prediction filter, M: # of mics,

m_0 : index of target channel to be dereverberated

α : a coeff. for flooring constant (e.g., $\alpha = 10^{-4}$)

2. Decompose a multichannel observed signal into a set of subband signals

$x_{n,f}^{(m)}$: subband signal (e.g., [Weiss 2000], or STFT can also be used)

m : channel index, n: sample index

f : subband index

E.g., # of subbands is 512 (including negative frequencies) for 16 kHz sampling

3. In each subband f, set initial estimates of source variance $\sigma_{n,f}$ as

$$\varepsilon_k = \alpha \max_n | x_{n,f}^{(m_0)} |^2$$

$$\sigma_{n,f} = \max\left\{ | x_{n,f}^{(m_0)} |^2, \varepsilon_f \right\}$$

where ε_f is a flooring constant for $\sigma_{n,f}$

4. Obtain vector representation of $x_{n,f}^{(m)}$ in all channels as

$$\mathbf{x}_{n,f} = [\mathbf{x}_{n,f}^{(1)\,T}, \mathbf{x}_{n,f}^{(2)\,T}, \cdots \mathbf{x}_{n,f}^{(M)T}]^T$$

where T is non-conjugate transposition, and

$$\mathbf{x}_{n,f}^{(m)} = [x_{n,f}^{(m)}, x_{n-1,f}^{(m)}, \cdots x_{n-L+1,f}^{(m)}]^T$$

5. In each subband f, iterate the following until convergence is achieved

i. Obtain prediction filter $\mathbf{c}_f$ as

$$\mathbf{c}_f = \left(\sum_n \frac{\mathbf{x}_{n-D,f} \mathbf{x}_{n-D,f}^{*T}}{\sigma_{n,f}} \right)^+ \sum_n \frac{\mathbf{x}_{n-D,f} \left(x_{n,f}^{(m_0)} \right)^*}{\sigma_{n,f}}$$

where $+$ and $*$ are Moore-Penrose pseudo-inverse and complex conjugate operations. (see [Yoshioka, 2009b] for efficient calculation)

ii. Obtain dereverberated subband signal $\mathbf{y}_{n,f}$ as

$$\mathbf{y}_{n,f} = x_{n,f}^{(m_0)} - \mathbf{c}_f^{*T} \mathbf{x}_{n-D,f}$$

iii. Update source variance estimates $\sigma_{n,f}$ as

$$\sigma_{n,f} = \max\left\{ | y_{n,f} |^2, \varepsilon_f \right\}$$

6. Compose a dereverberated signal from a set of dereverberated subband signals $\mathbf{y}_{n,f}$

Part II. Multichannel blind inverse filtering

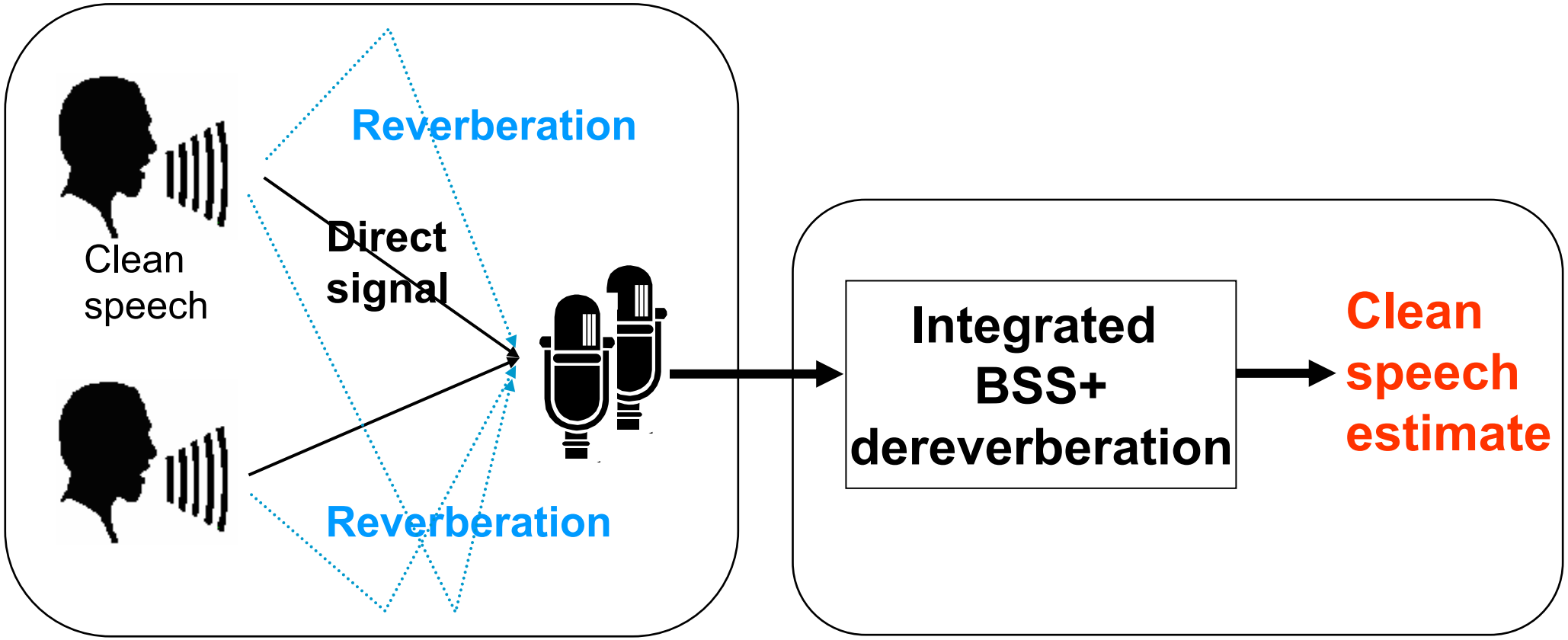
- Example applications
  - Professional audio post production
  - Meeting recognition with microphone arrays

- Fundamentals: dereverberation with inverse filtering
  - What is inverse filter
  - Robust ‘approximate’ inverse filter

- Blind inverse filtering
  - Overview of basic approaches
  - Closer look: multichannel linear prediction with time-varying source model

- **Integration with blind source separation**

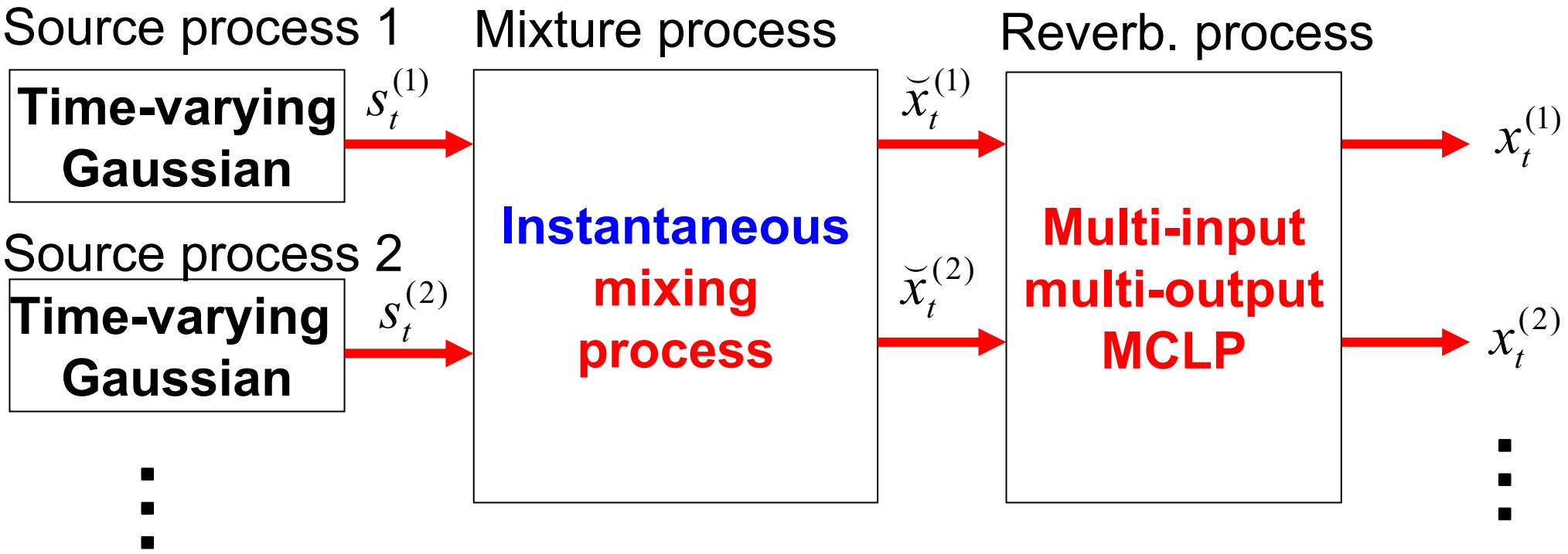
BSS+dereverberation



Approaches:

- MCLP based approach [Yoshioka 2009b, 2011]
- TRINICON [Buchner 2010]

Generative model for reverberant sound mixture

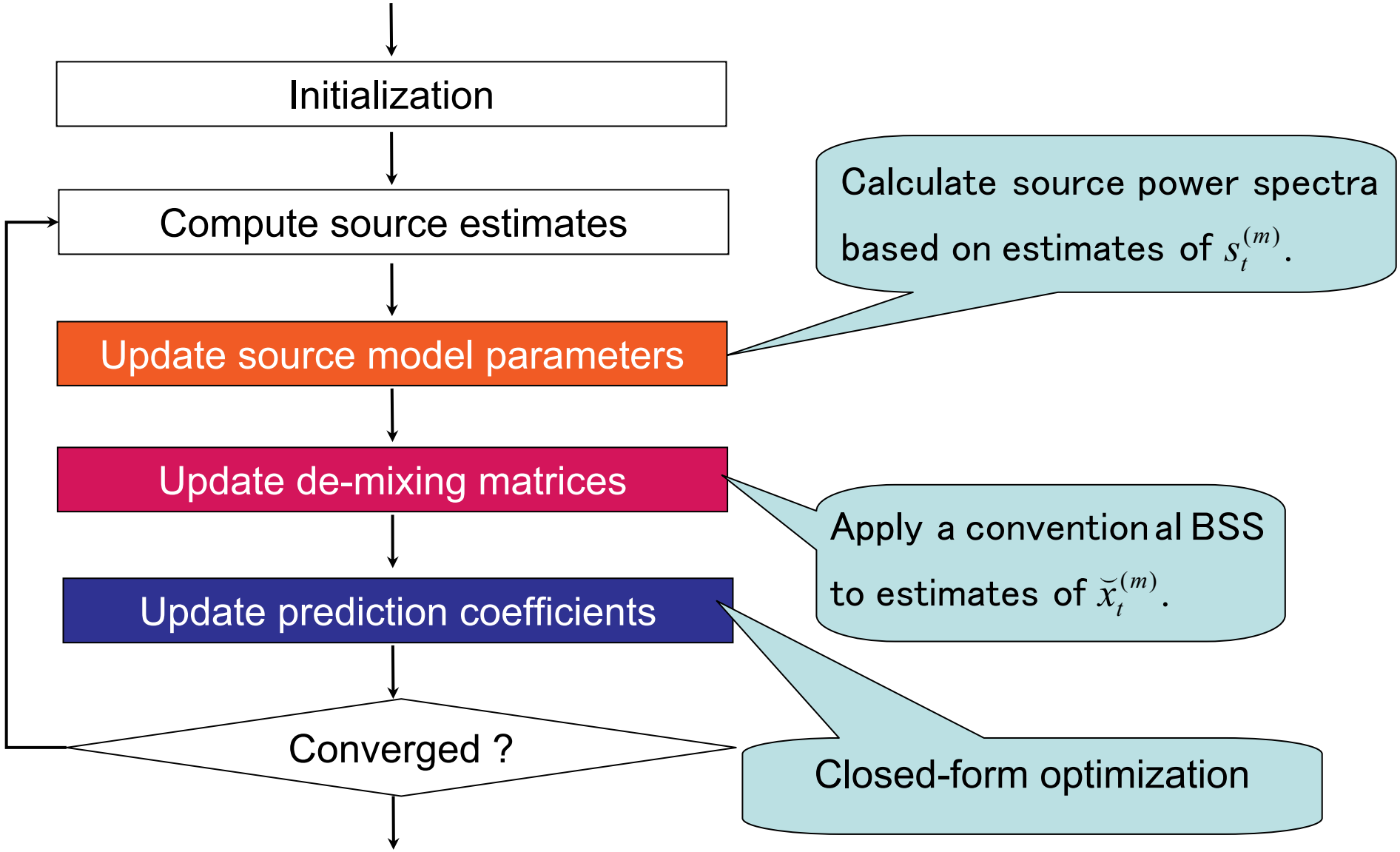


$\breve{x}_t^{(m)}$: non-reverberant mixture

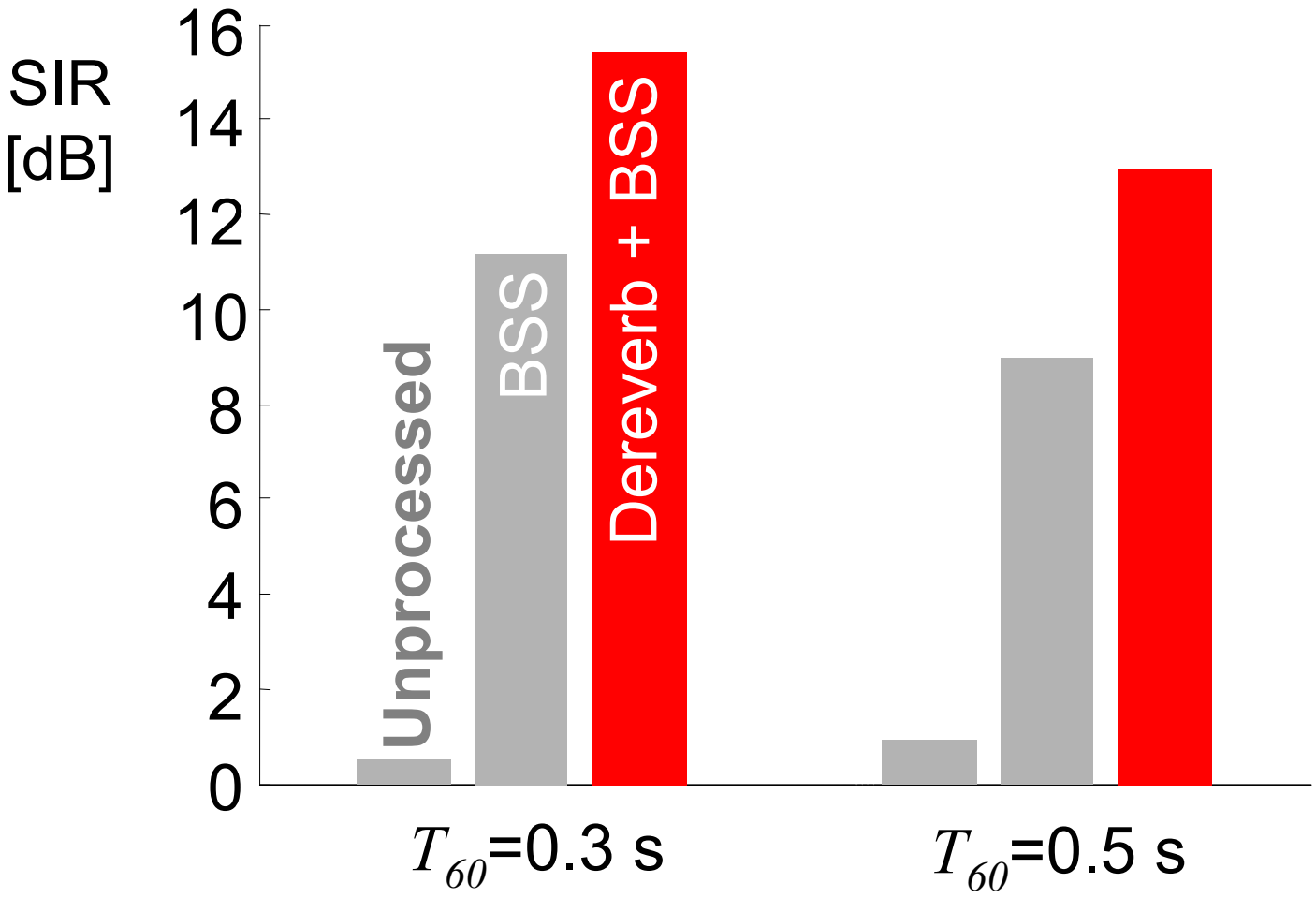
$x_t^{(m)}$: reverberant mixture

Jointly optimized by maximum likelihood estimation approach [Yoshioka 2009b, 2011]

Optimization procedure (subband-based implementation)



Improvement in signal-to-interference ratio (SIR)



sources: 2
mics: 4
Source-mic distance: 1.5 m

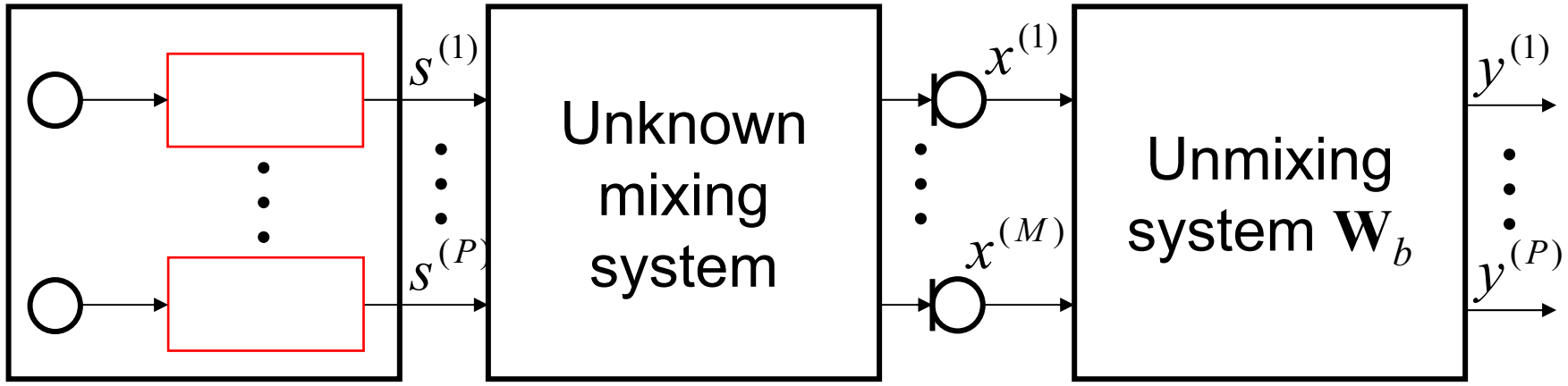
Recording: 1 to 8 s (average: 3.5 s)

BSS: [Sawada 2007]

Results averaged over 672 pairs of utterances (TIMIT test set)

Live demo

TRINICON: general framework for blind MIMO signal processing



Cost function [Buchner 2010]

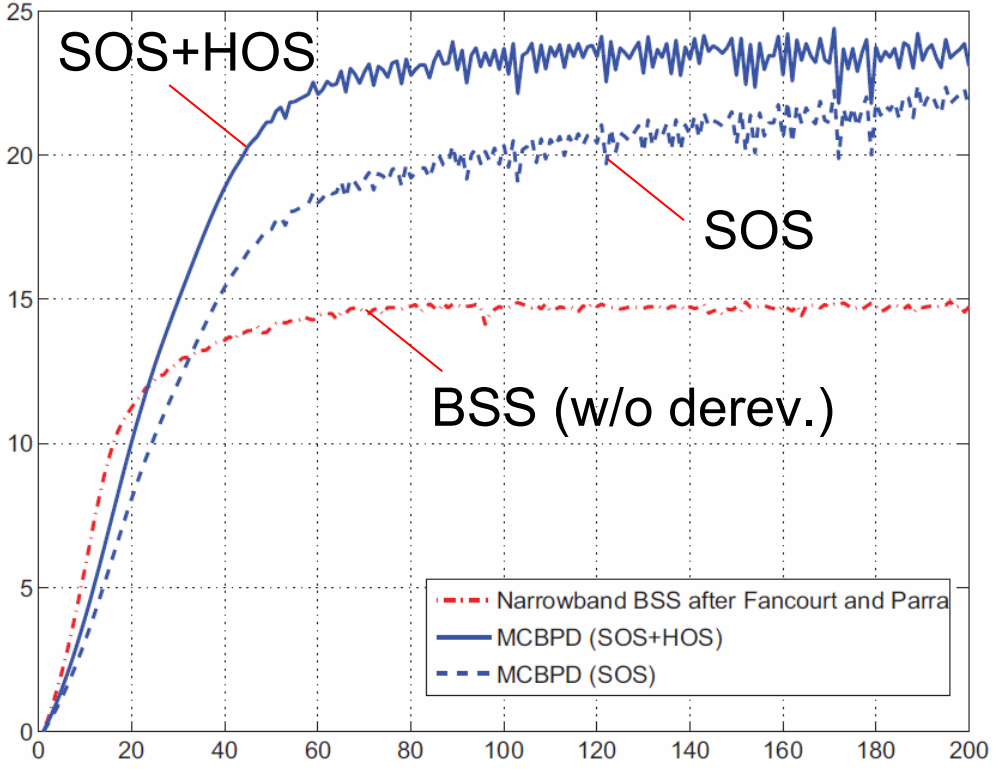
$$\mathcal{J}(\mathbf{W}_b) = -\sum_{i=0}^{\infty} \beta(i,b) \sum_{j=0}^{N_0} \left\{ \log(\hat{p}_{s,PD}(\mathbf{y}(i,j))) - \log(\hat{p}_{y,PD}(\mathbf{y}(i,j))) \right\}$$

with PD-variate pdfs (P: source number, D: filter length)

- $\hat{p}_{s,PD}(\mathbf{y}(i,j))$ for source (assumed or estimated)
- $\hat{p}_{y,PD}(\mathbf{y}(i,j))$ for output

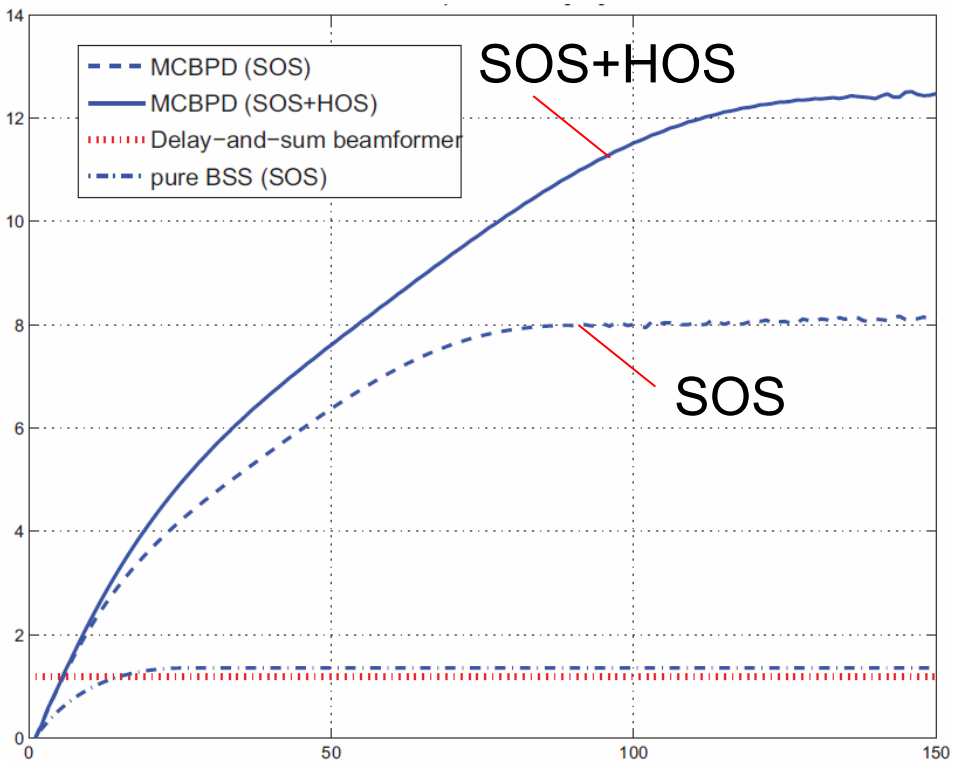
Comparison of SOS and HOS by TRINICON [Buchner 2010]

SIR improvement (dB)



Number of iterations

Signal-to-reverberation ratio (SRR) improvement (dB)



Number of iterations

mics.: 4, # sources: 2, T_{60}: 700 ms,
Source-mic distance: 1.65 m, Recording: 30 sec

Summary II-2

- Robust blind inverse filtering is possible
  - Using joint speech and reverberation modeling
    - Based only on a few seconds of observation (e.g., 2.5 s)
    - With a relatively small computational cost (e.g., RTF<1)
    - In an online processing manner (e.g., latency=1s)
  - Under low SNR conditions (e.g., 10 dB SNR)

- Future challenges
  - Realtime adaptation of inverse filter [Yoshioka 2009a], [Evers 2011]
  - Single channel inverse filtering [Gillespie 2001]
  - Processing under more adverse noise conditions such as nonstationary diffuse noise
  - Optimal integration of inverse filtering and spectral enhancement based dereverberation

Part III:

Robust Automatic Speech Recognition (ASR) in Reverberant Environments

Part III: Robust ASR in Reverberant Environments

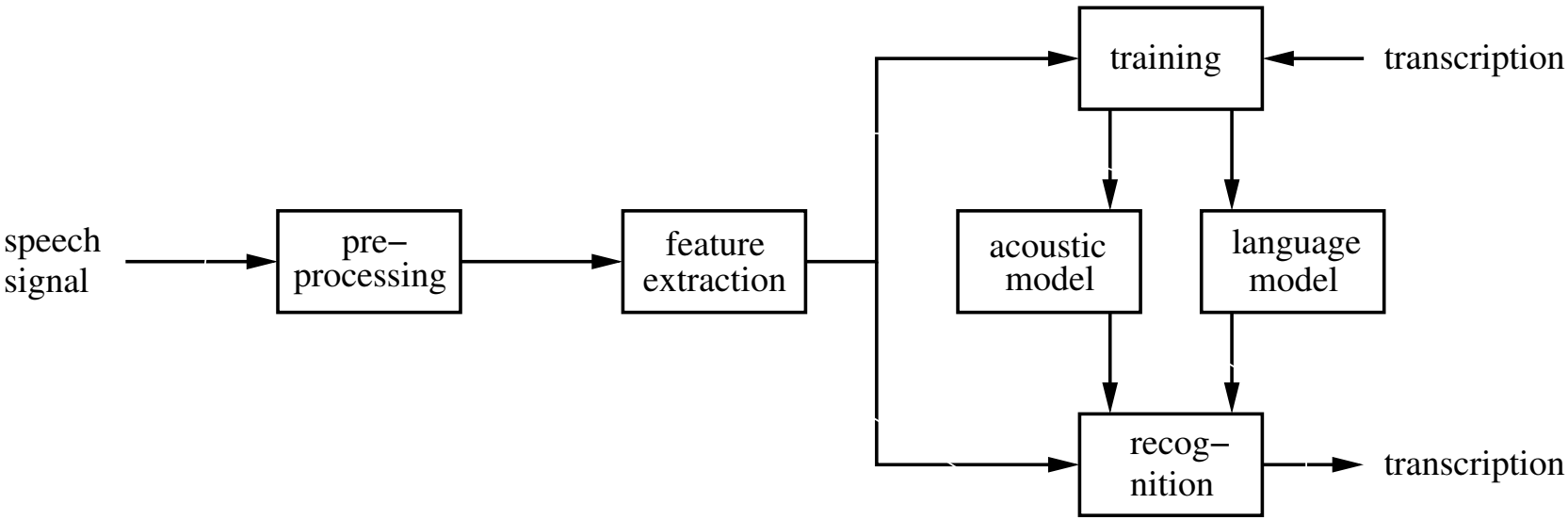
- Introduction
- Feature-based Approaches
- Model-based Approaches
- Decoder-based Approaches
- REMOS

Part III: Robust ASR in Reverberant Environments

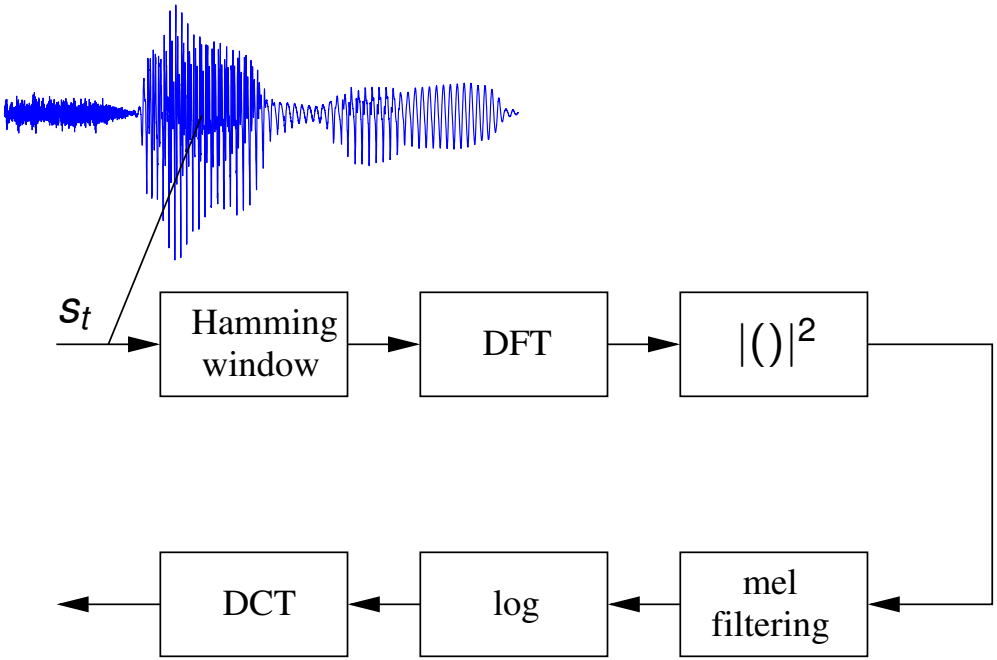
- Introduction
- Feature-based Approaches
- Model-based Approaches
- Decoder-based Approaches
- REMOS

ASR System

Block Diagram



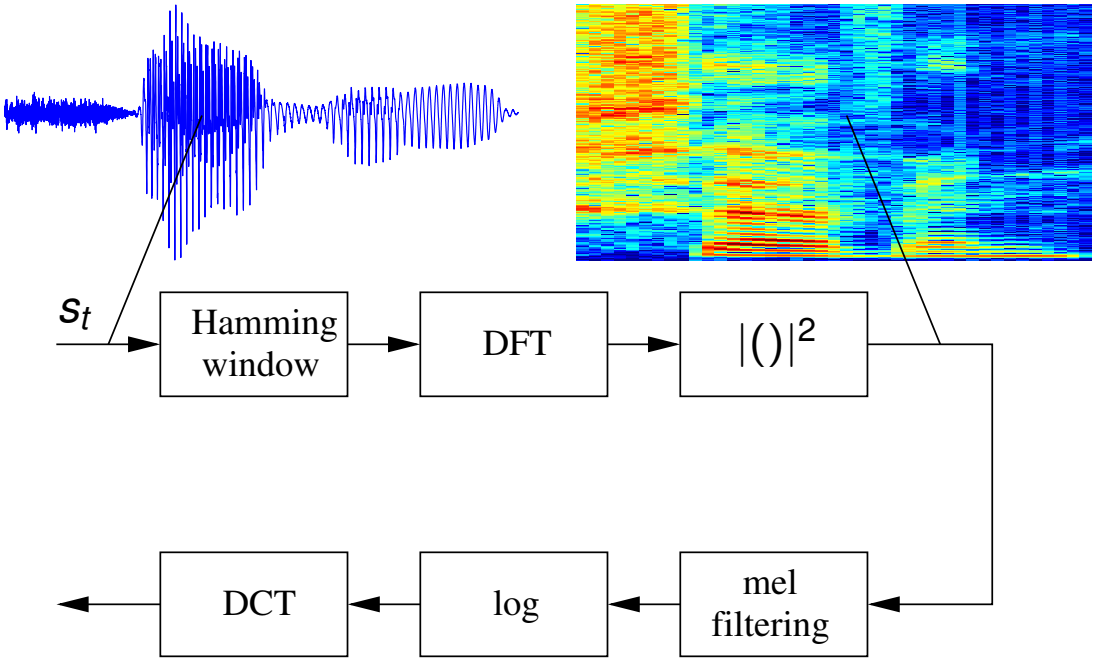
Feature Extraction: Calculation of MFCCs



Goal:
Dimensionality reduction

MFCCs:
Mel Frequency Cepstral Coefficients

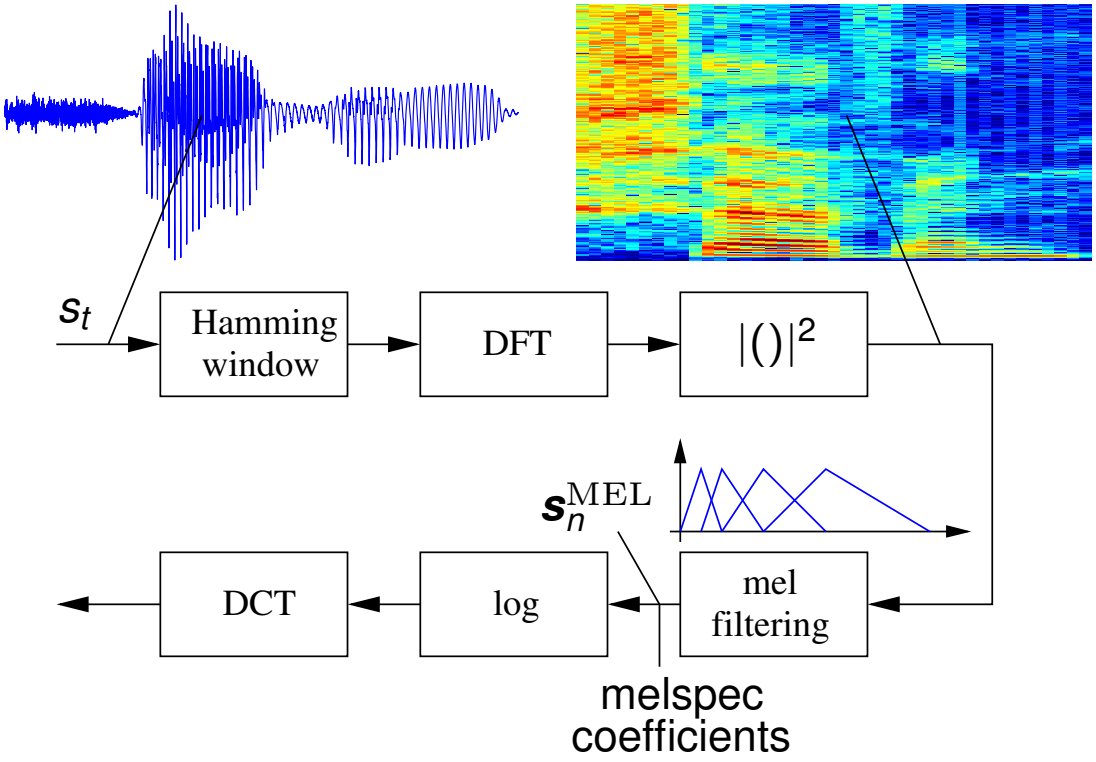
Feature Extraction: Calculation of MFCCs



Goal:
Dimensionality reduction

MFCCs:
Mel Frequency Cepstral Coefficients

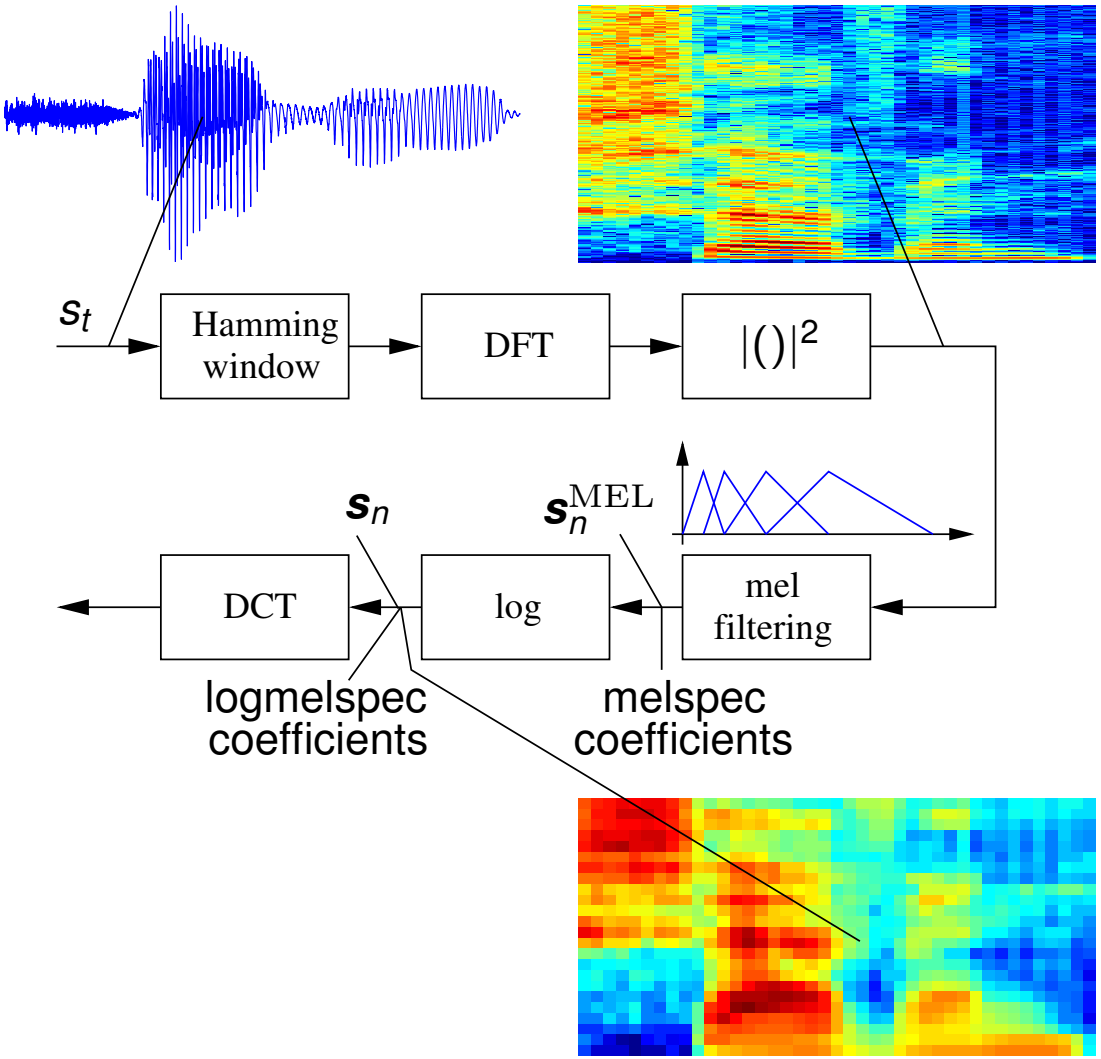
Feature Extraction: Calculation of MFCCs



Goal:
Dimensionality reduction

MFCCs:
Mel Frequency Cepstral Coefficients

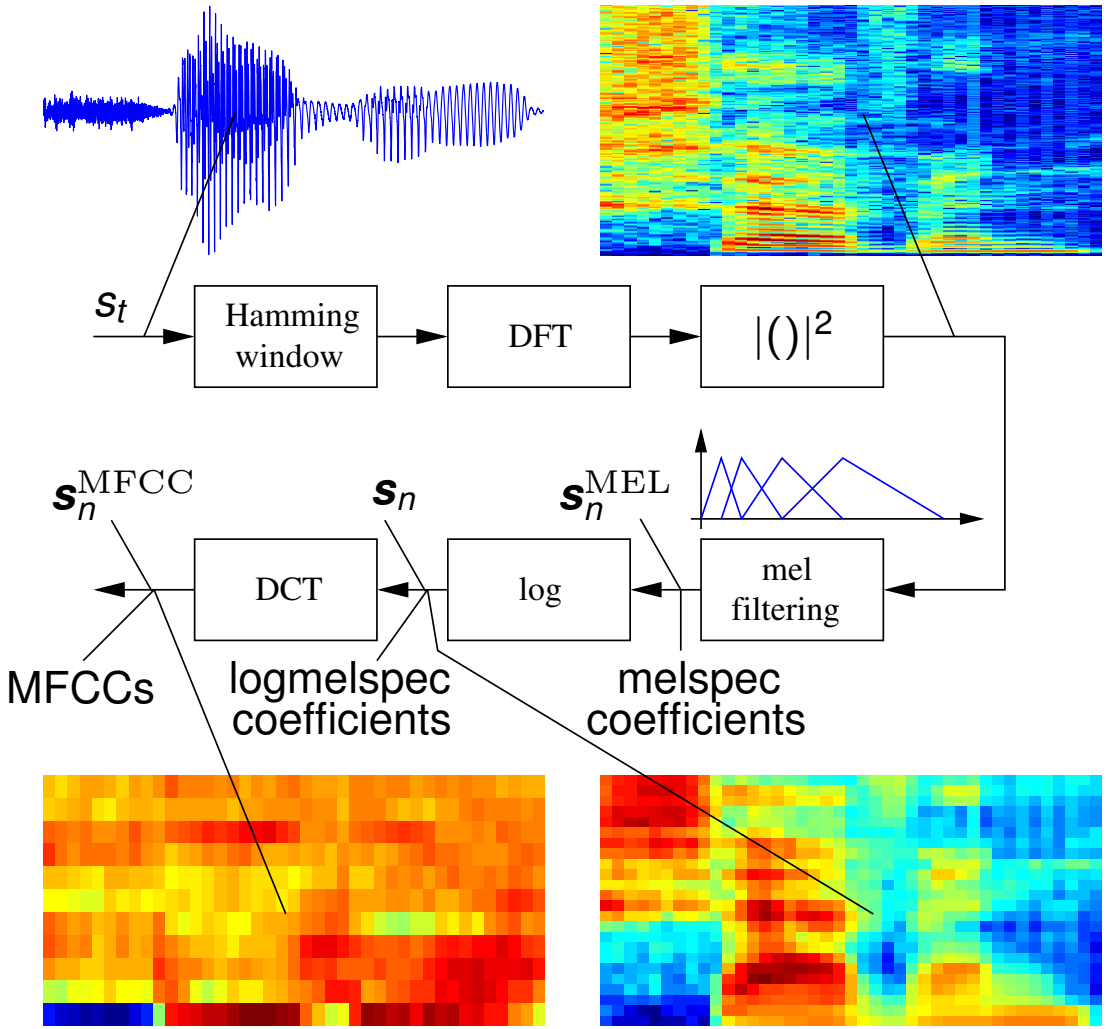
Feature Extraction: Calculation of MFCCs



Goal:
Dimensionality reduction

MFCCs:
Mel Frequency Cepstral Coefficients

Feature Extraction: Calculation of MFCCs

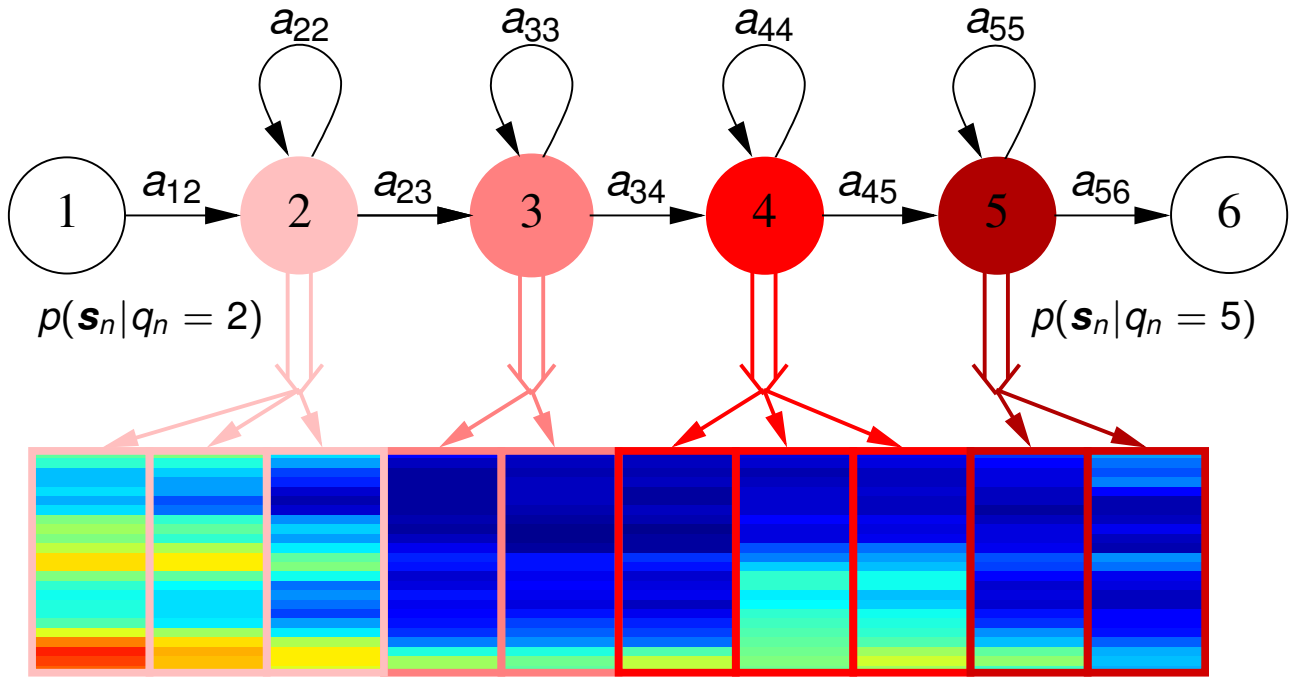


Goal:
Dimensionality reduction

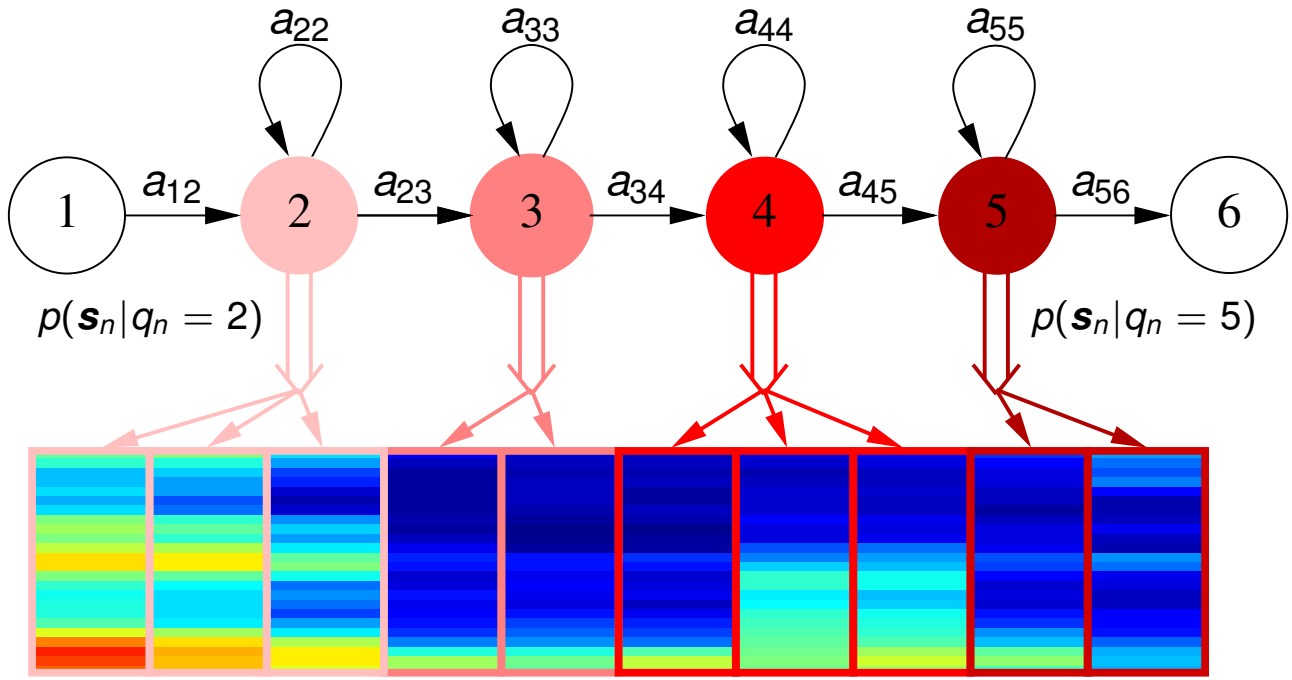
MFCCs:
Mel Frequency Cepstral Coefficients

Acoustic Modeling

Hidden Markov Model (HMM) λ



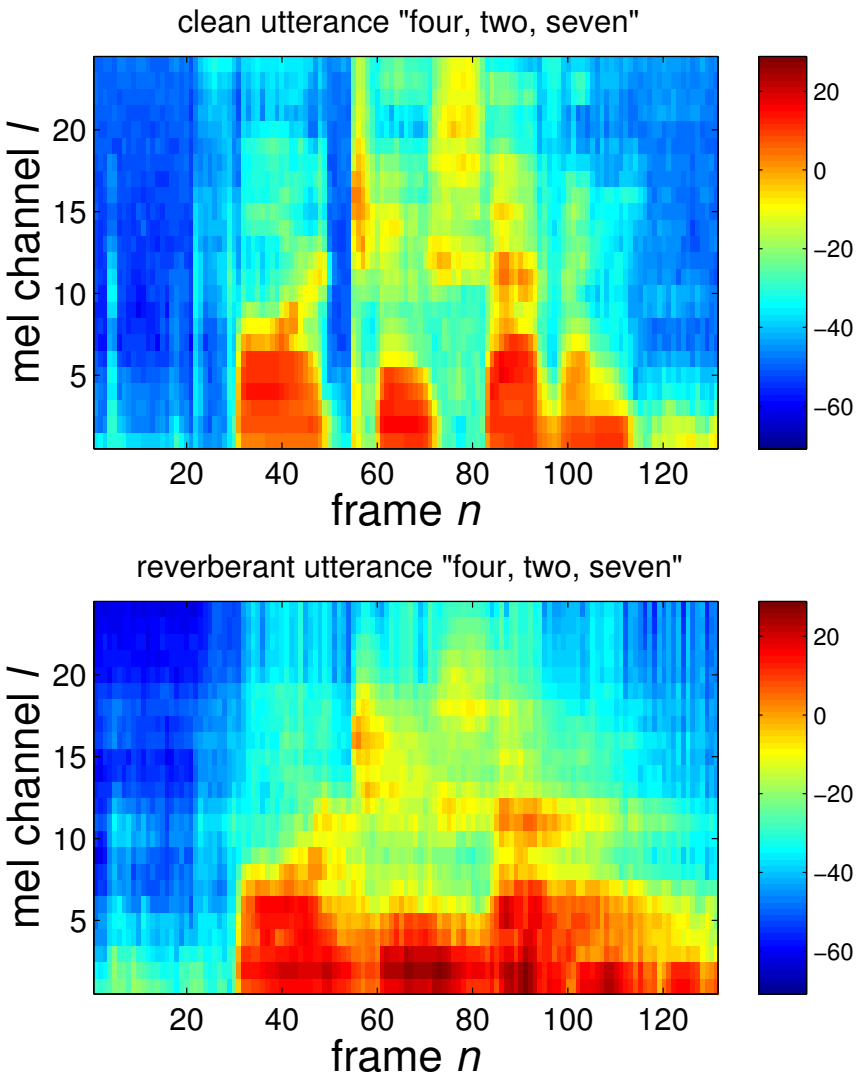
Hidden Markov Model (HMM) λ



Powerful model for:

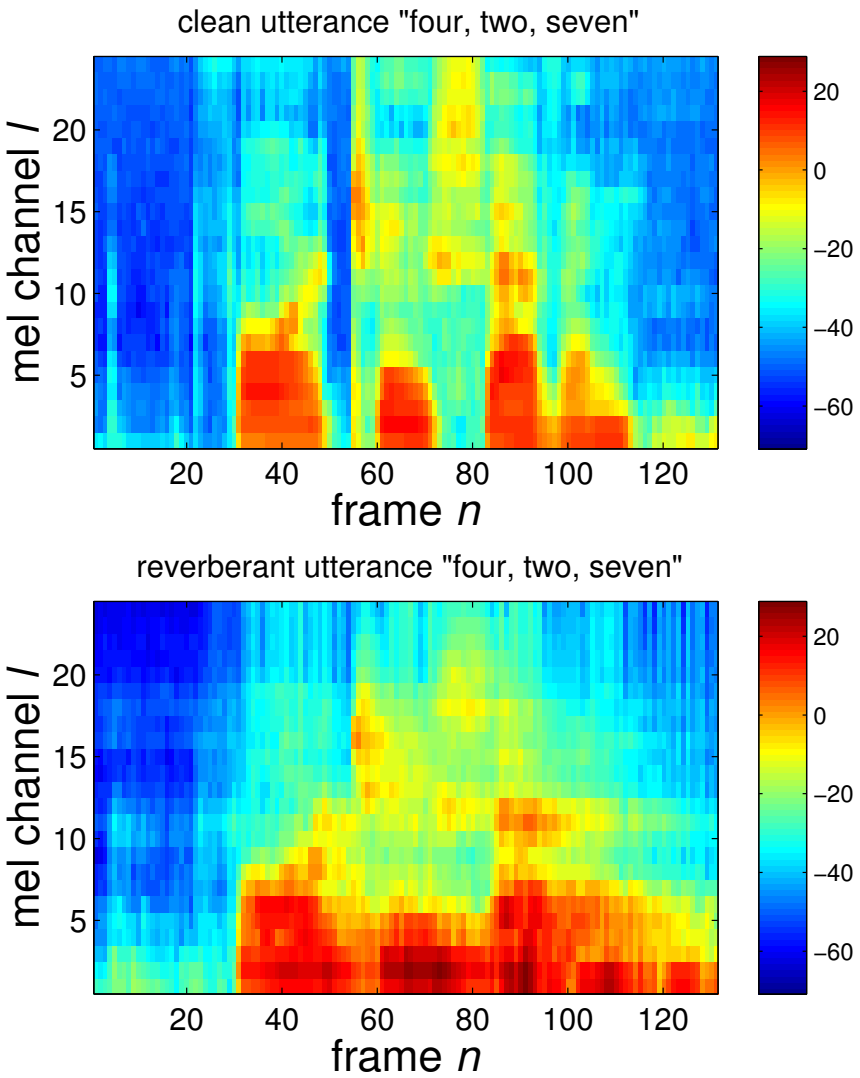
- temporal variation
- spectral variation

Dispersive Effect of Reverberation



- Logmelspec features, dB scale
- Dispersive effect of reverberation: features smeared along time axis

Dispersive Effect of Reverberation

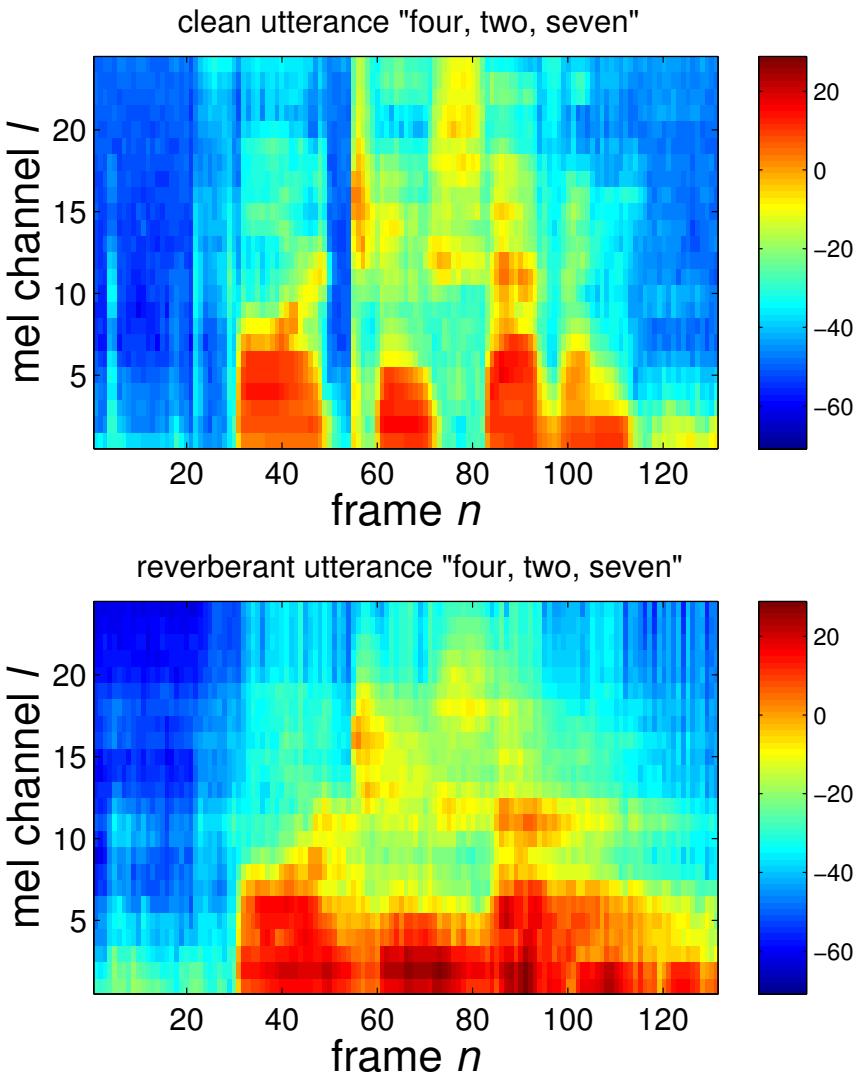


- Logmelspec features, dB scale
- Dispersive effect of reverberation: features smeared along time axis

⇩

- Time-frequency pattern is changed
- Inter-frame correlation is increased

Dispersive Effect of Reverberation



- Logmelspec features, dB scale
- Dispersive effect of reverberation: features smeared along time axis

⇩

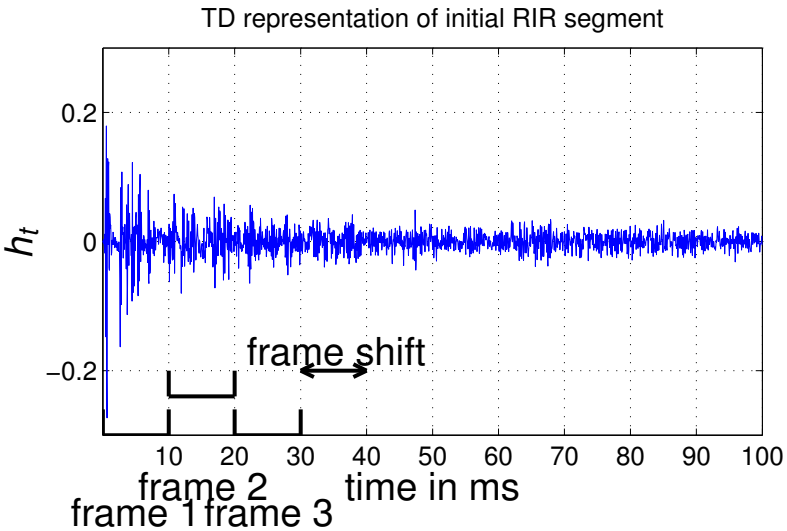
- Time-frequency pattern is changed
- Inter-frame correlation is increased

⇩

Different statistical properties to be captured by acoustic model

Contradiction to conditional inde-pendence assumption of HMMs

Explanation of Dispersive Effect

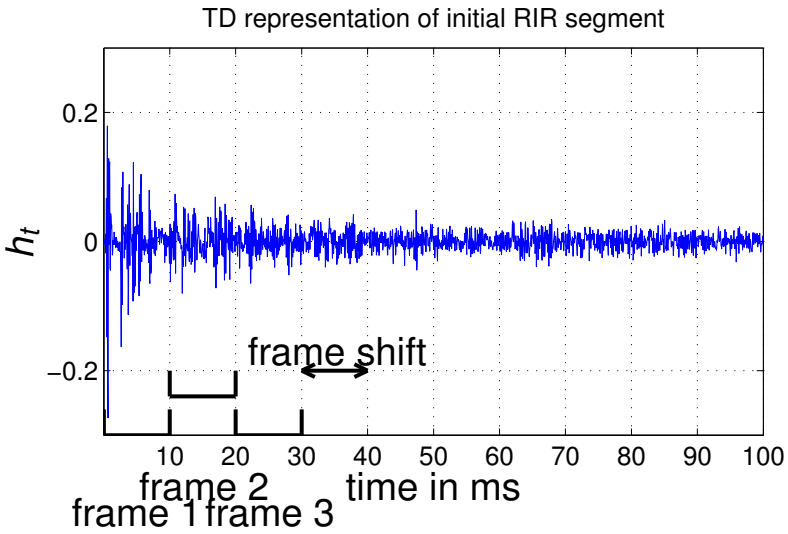


- **Time-domain (TD) description of reverberant speech x_t:**

  $x_t = h_t * s_t$

  RIR typically much longer than analysis window

Explanation of Dispersive Effect



- **Time-domain (TD) description of reverberant speech x_t:**

  $x_t = h_t * s_t$

  RIR typically much longer than analysis window

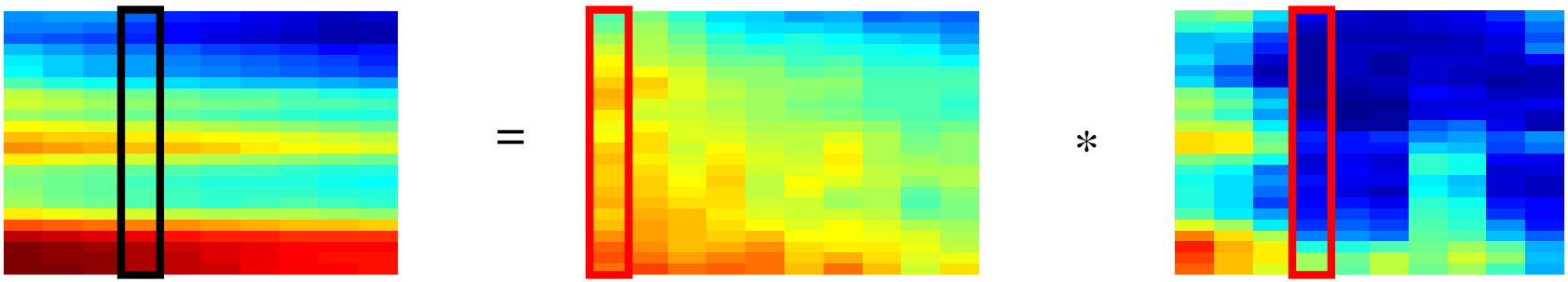
- **Feature-domain (FD) description of $\boldsymbol{x}_n^{\mathrm{MEL}}$: melspec convolution**

$$\boldsymbol{x}_n^{\mathrm{MEL}} = \sum_{\tau=0}^{T_H-1} \boldsymbol{h}_\tau^{\mathrm{MEL}} \odot \boldsymbol{s}_{n-\tau}^{\mathrm{MEL}}$$

$\boldsymbol{s}_n^{\mathrm{MEL}}$: clean-speech feature vector
$\boldsymbol{x}_n^{\mathrm{MEL}}$: reverberant feature vector
$\boldsymbol{h}_n^{\mathrm{MEL}}$: melspec RIR representation
$\odot$: element-wise multiplication

Illustration of Melspec Convolution

$$\boldsymbol{x}_n^{\mathrm{MEL}} = \boldsymbol{h}_n^{\mathrm{MEL}} * \boldsymbol{s}_n^{\mathrm{MEL}}$$



$$\boldsymbol{x}_n^{\mathrm{MEL}} = \boldsymbol{h}_0^{\mathrm{MEL}} \odot \boldsymbol{s}_n^{\mathrm{MEL}}$$

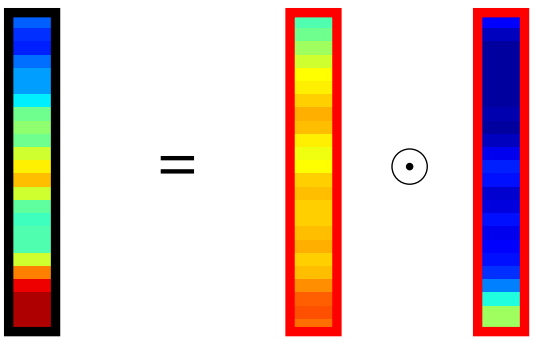
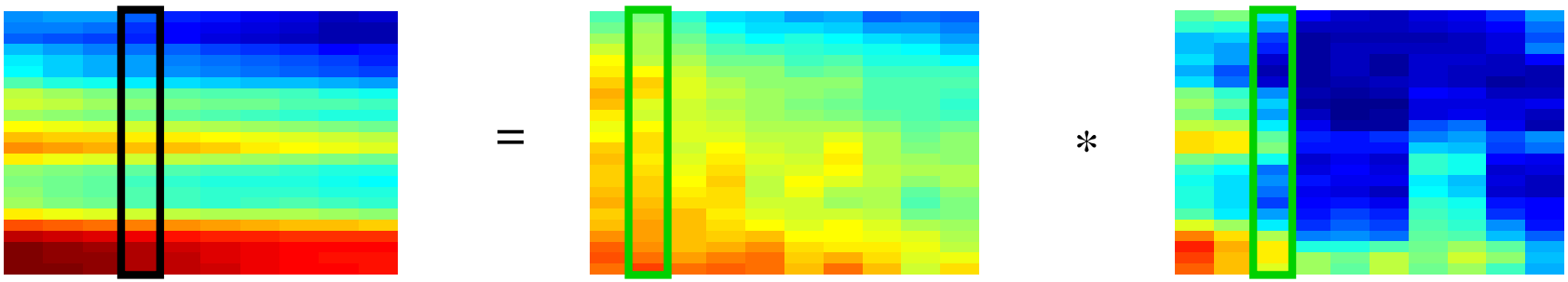


Illustration of Melspec Convolution

$$\boldsymbol{x}_n^{\mathrm{MEL}} = \boldsymbol{h}_n^{\mathrm{MEL}} * \boldsymbol{s}_n^{\mathrm{MEL}}$$



$$\boldsymbol{x}_n^{\mathrm{MEL}} = \boldsymbol{h}_0^{\mathrm{MEL}} \odot \boldsymbol{s}_n^{\mathrm{MEL}} + \boldsymbol{h}_1^{\mathrm{MEL}} \odot \boldsymbol{s}_{n-1}^{\mathrm{MEL}}$$

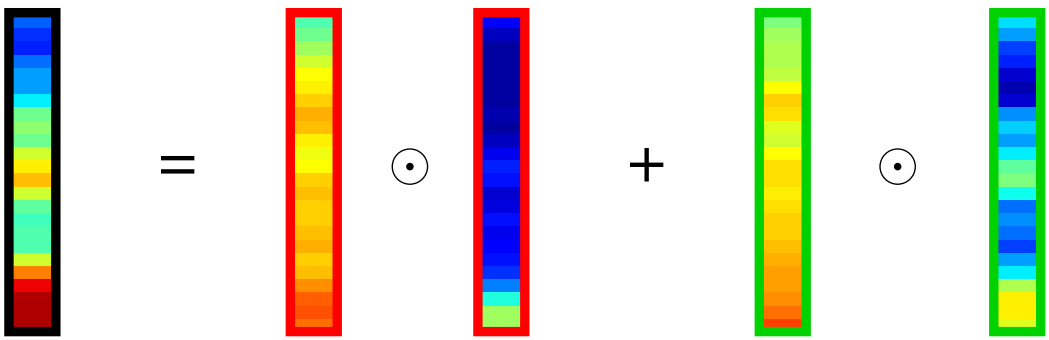
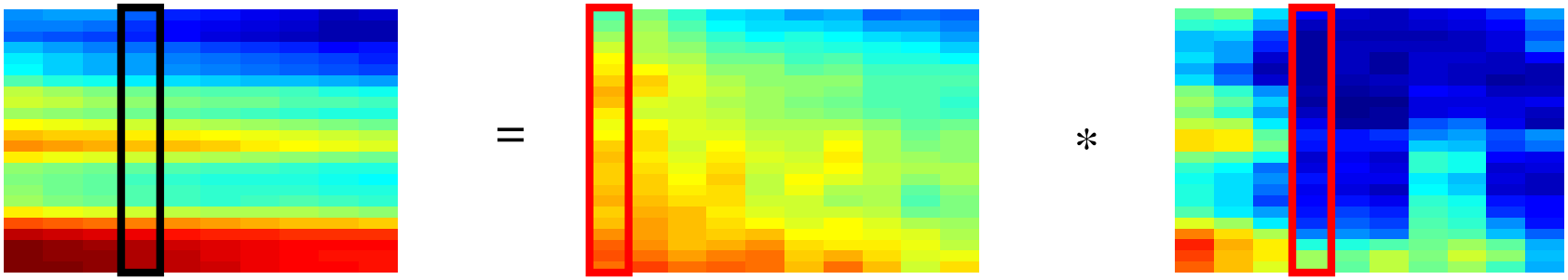
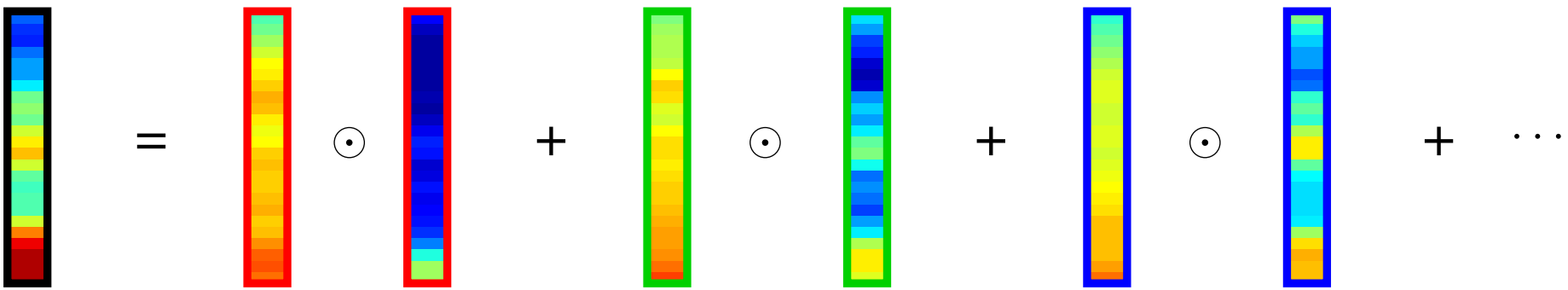


Illustration of Melspec Convolution

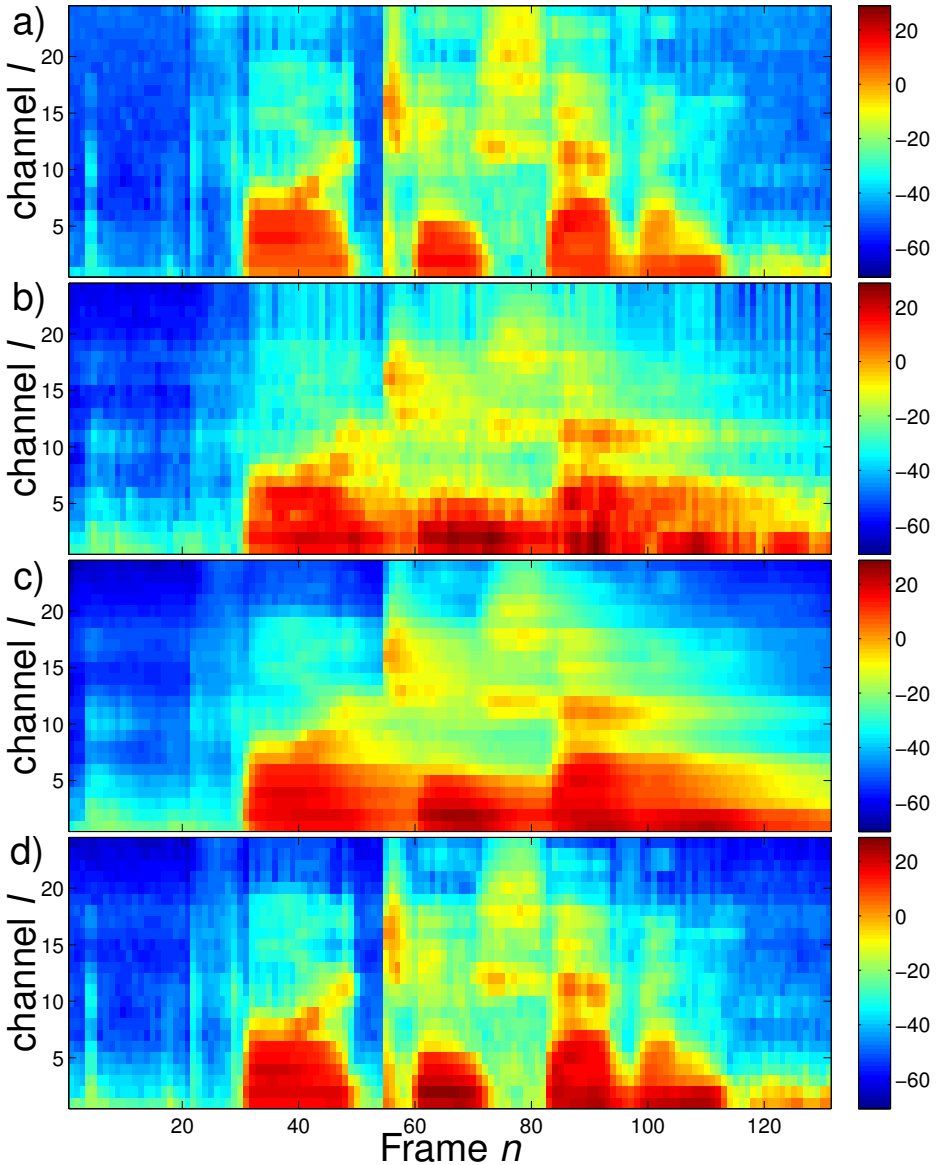
$$\boldsymbol{x}_n^{\mathrm{MEL}} = \boldsymbol{h}_n^{\mathrm{MEL}} * \boldsymbol{s}_n^{\mathrm{MEL}}$$



$$\boldsymbol{x}_n^{\mathrm{MEL}} = \boldsymbol{h}_0^{\mathrm{MEL}} \odot \boldsymbol{s}_n^{\mathrm{MEL}} + \boldsymbol{h}_1^{\mathrm{MEL}} \odot \boldsymbol{s}_{n-1}^{\mathrm{MEL}} + \boldsymbol{h}_2^{\mathrm{MEL}} \odot \boldsymbol{s}_{n-2}^{\mathrm{MEL}} + \cdots$$



Accuracy of Melspec Convolution



a) Clean utterance

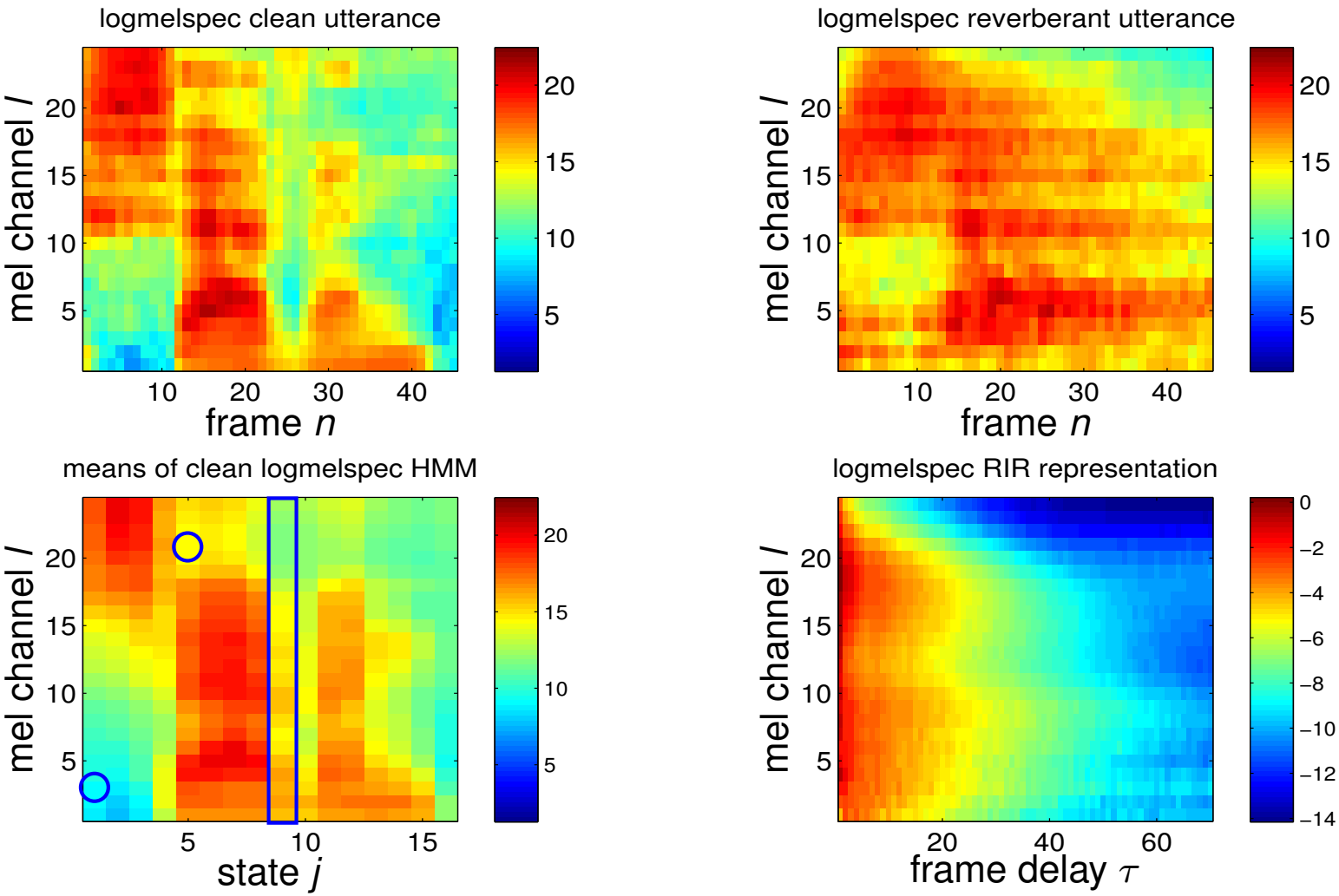
b) Reverberant utterance

c) Melspec convolution

d) Simple multiplication

Statistical Properties of Reverberant Speech Features

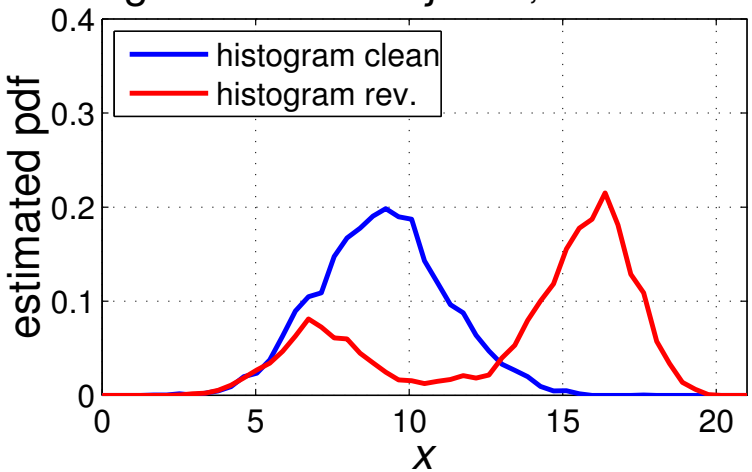
Example: digit “seven”



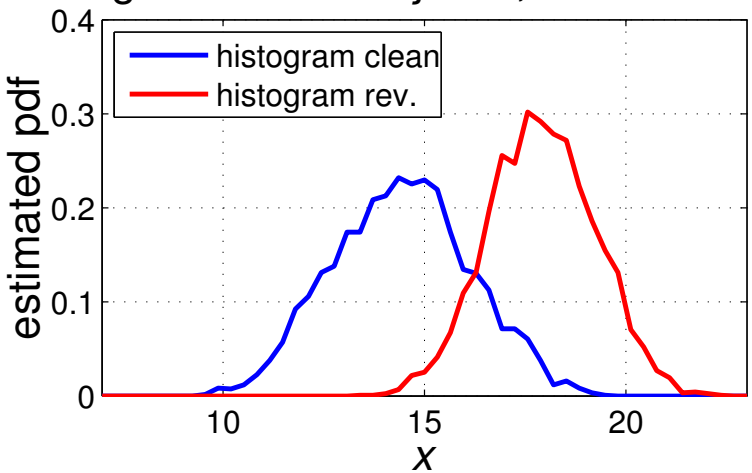
Statistical Properties of Reverberant Speech Features

Histograms

histograms for state $j = 1$, channel $l = 3$

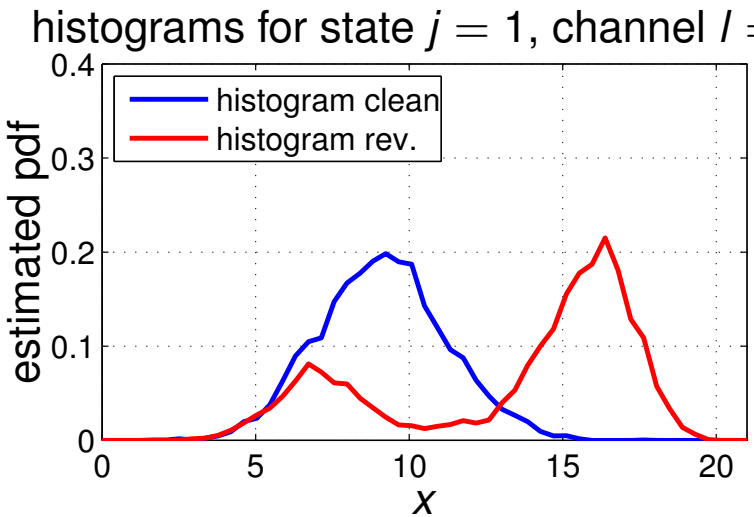


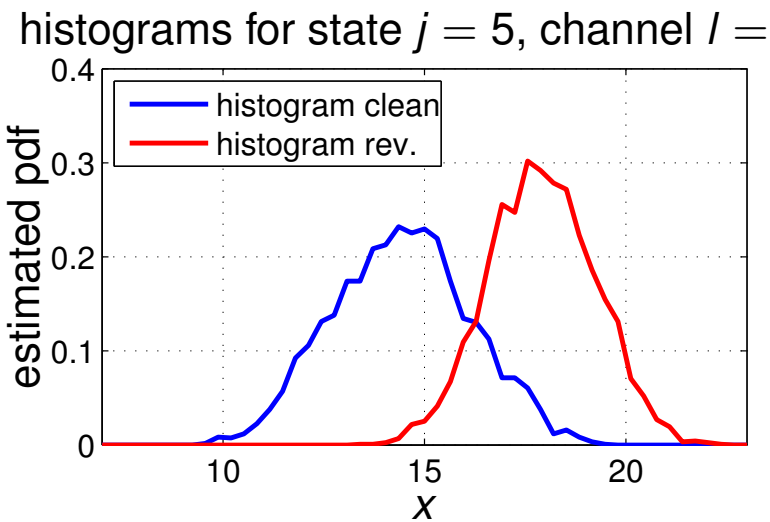
histograms for state $j = 5$, channel $l = 21$



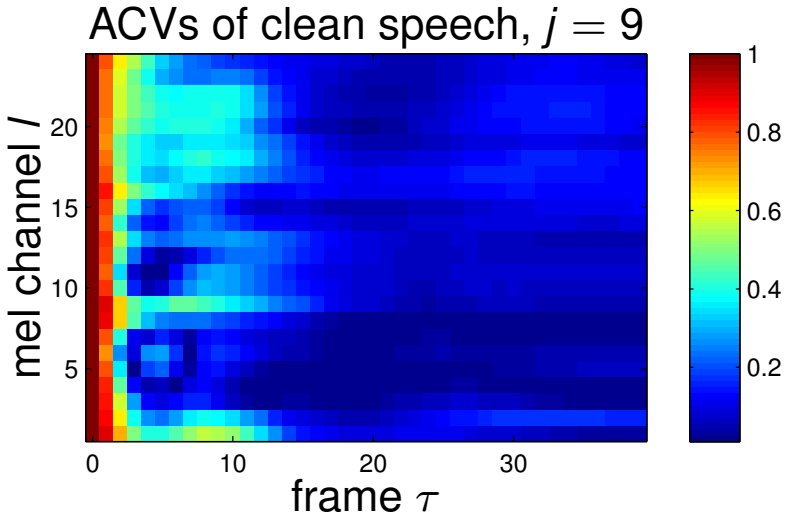
Statistical Properties of Reverberant Speech Features

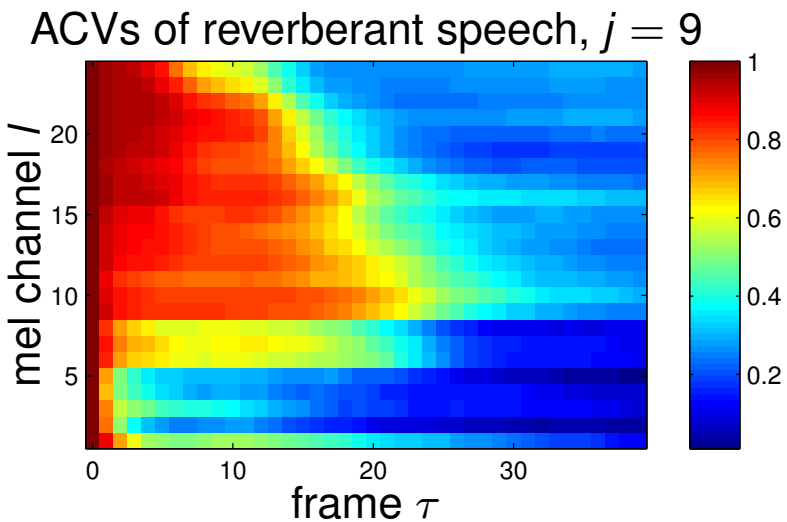
Histograms





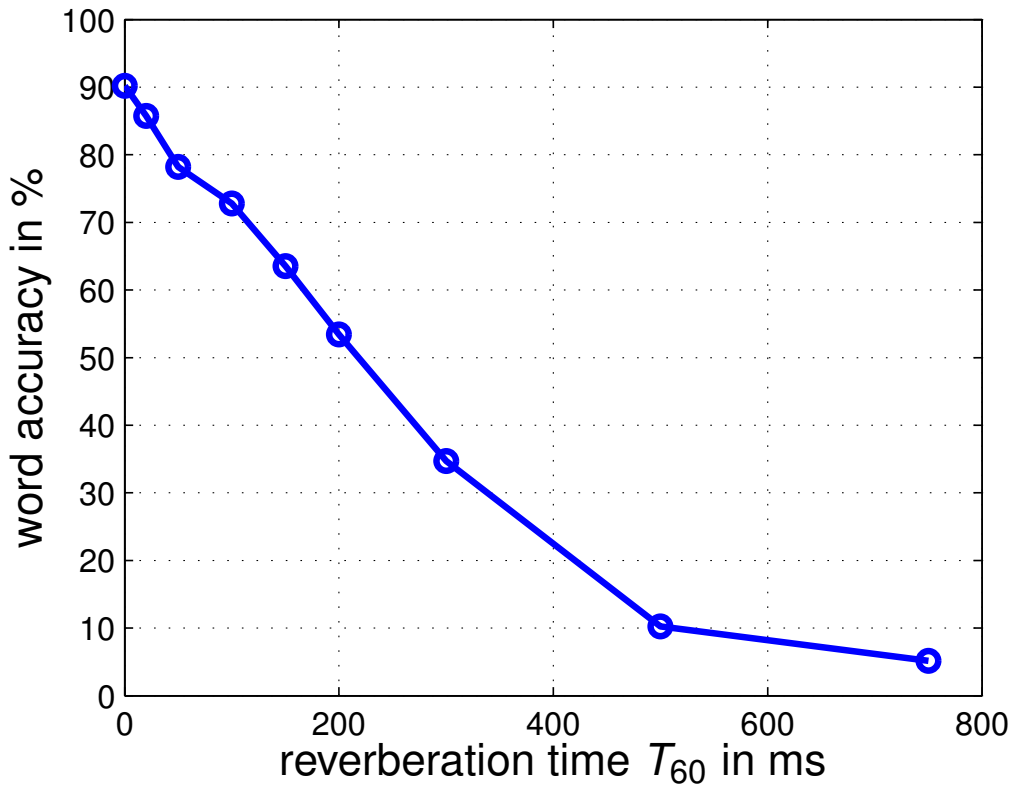
Auto-CoVariances (ACVs)





Recognition Results

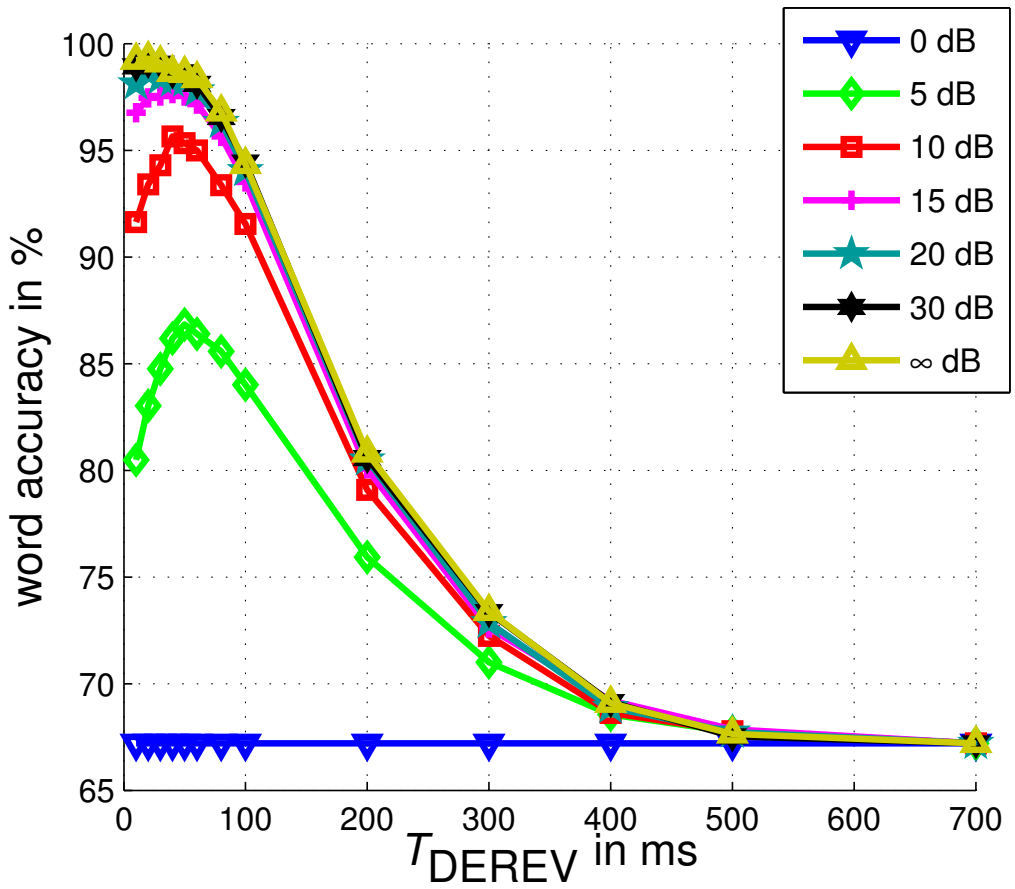
Word Accuracy as Function of Reverberation Time



- **Task:** Read sentences from Wall Street Journal (WSJ 5K task)
- **Features:** MFCCs + Δ + ΔΔ coefficients
- **Recognizer:** Cross-word triphones, 3 states per triphone, 16 Gaussians per state

Which Part of Reverberation is Harmful for ASR?

Word Accuracy as Function of Dereverberation Start Time

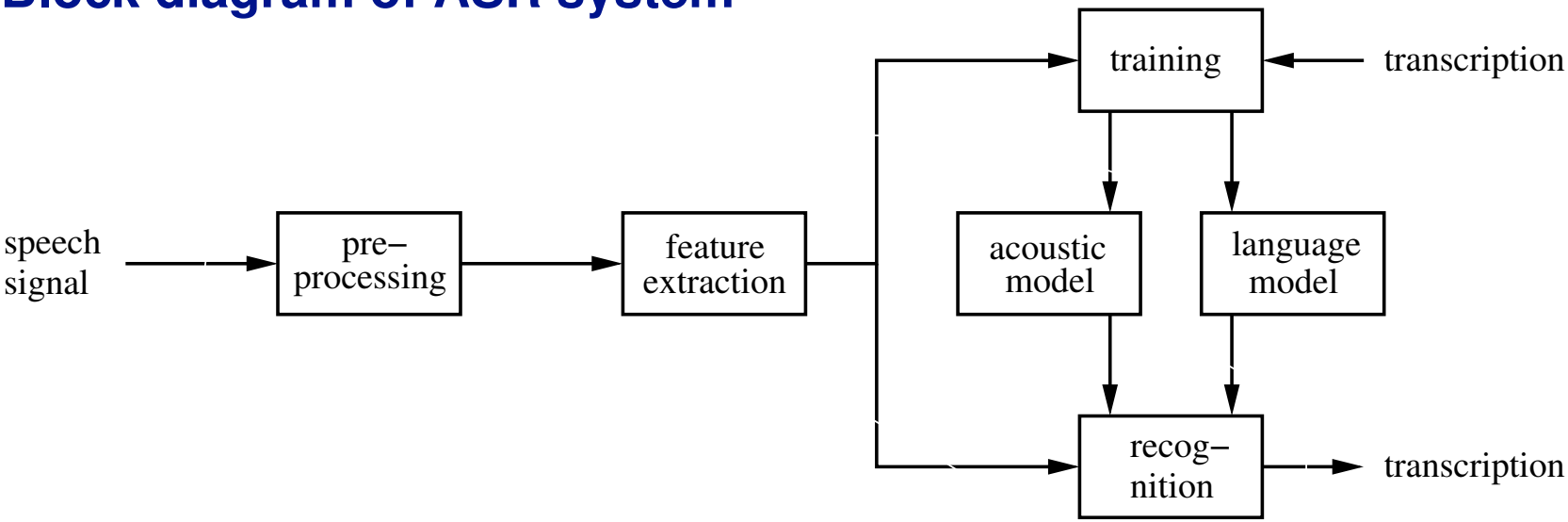


[Sehr 2010a]

- **Task:** Connected digits (TI digits)
- **Features:** MFCCs + Δ coefficients
- **Recognizer:** Word-level HMMs, 16 states per digit, 3 Gaussians per state

Strategies for Reverberation-Robust ASR

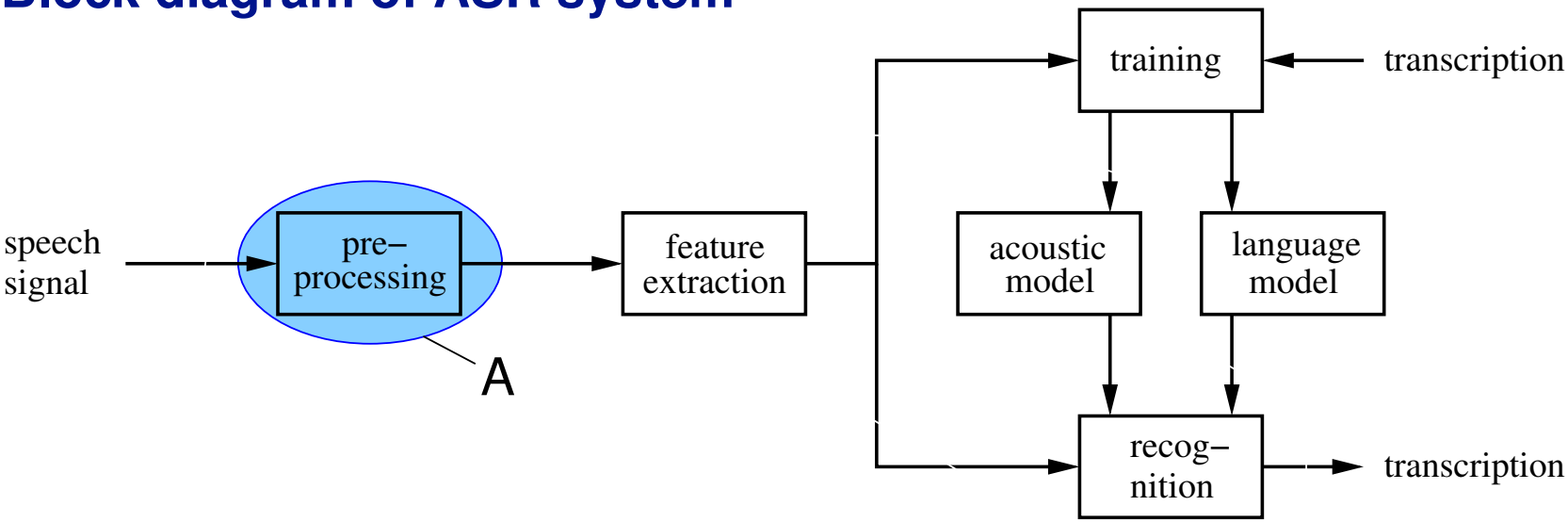
Block diagram of ASR system



Strategies

Strategies for Reverberation-Robust ASR

Block diagram of ASR system

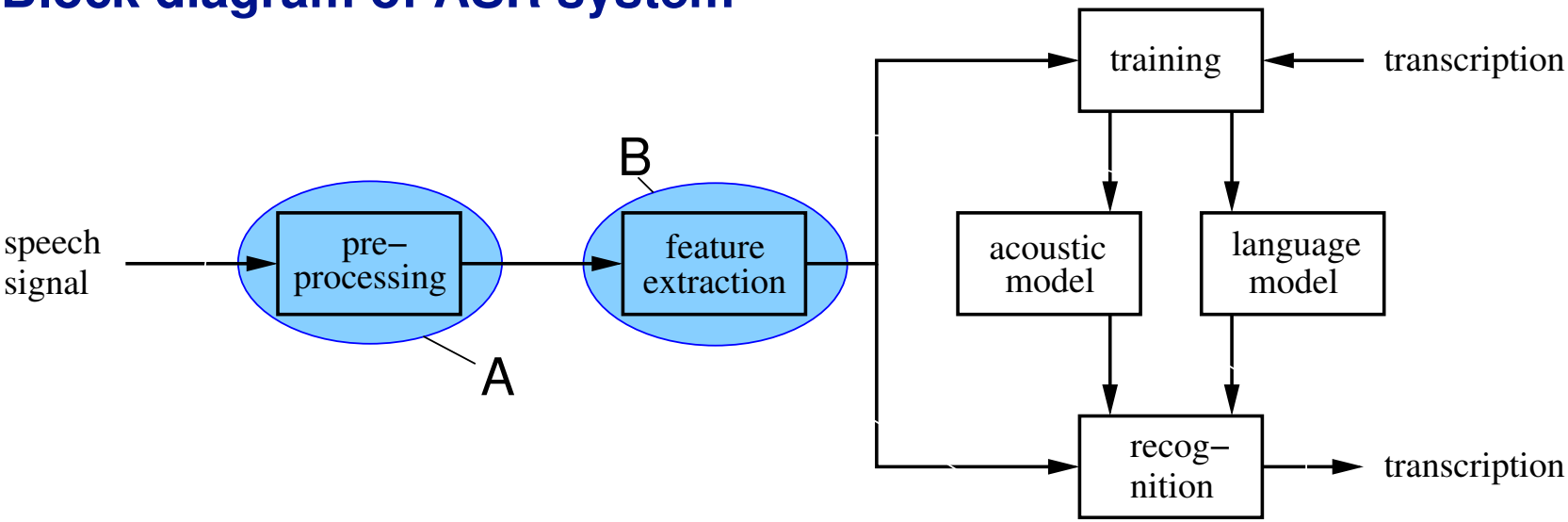


Strategies

A) signal-based approaches

Strategies for Reverberation-Robust ASR

Block diagram of ASR system



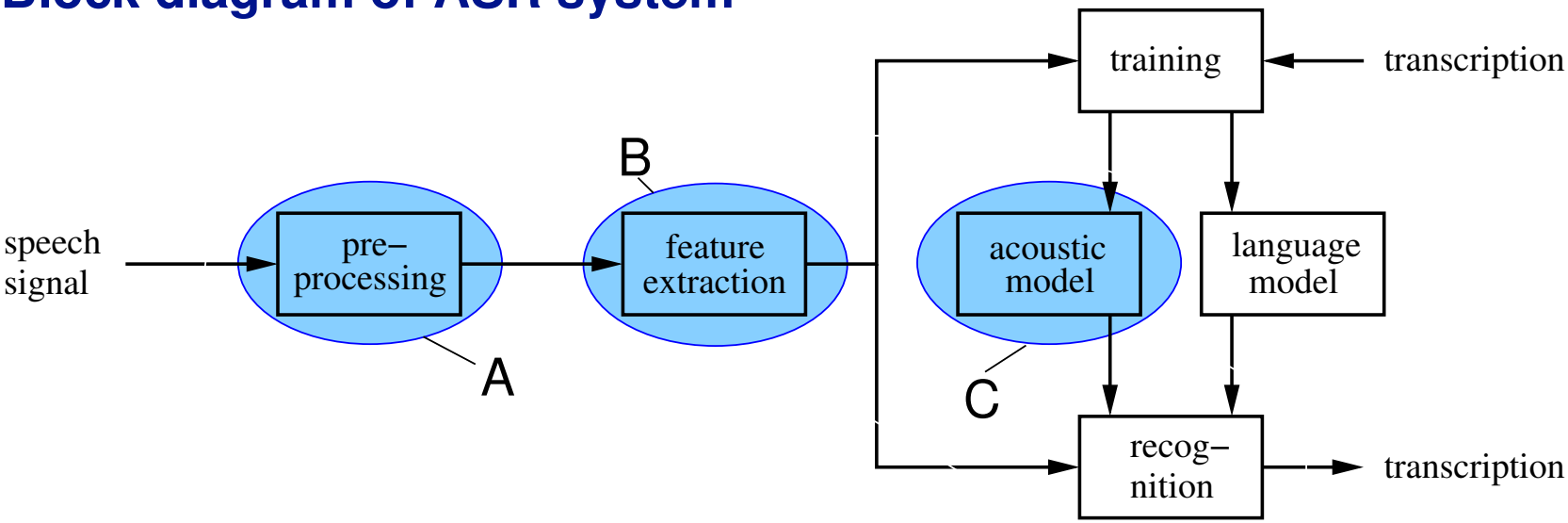
Strategies

A) signal-based approaches

B) feature-based approaches

Strategies for Reverberation-Robust ASR

Block diagram of ASR system



Strategies

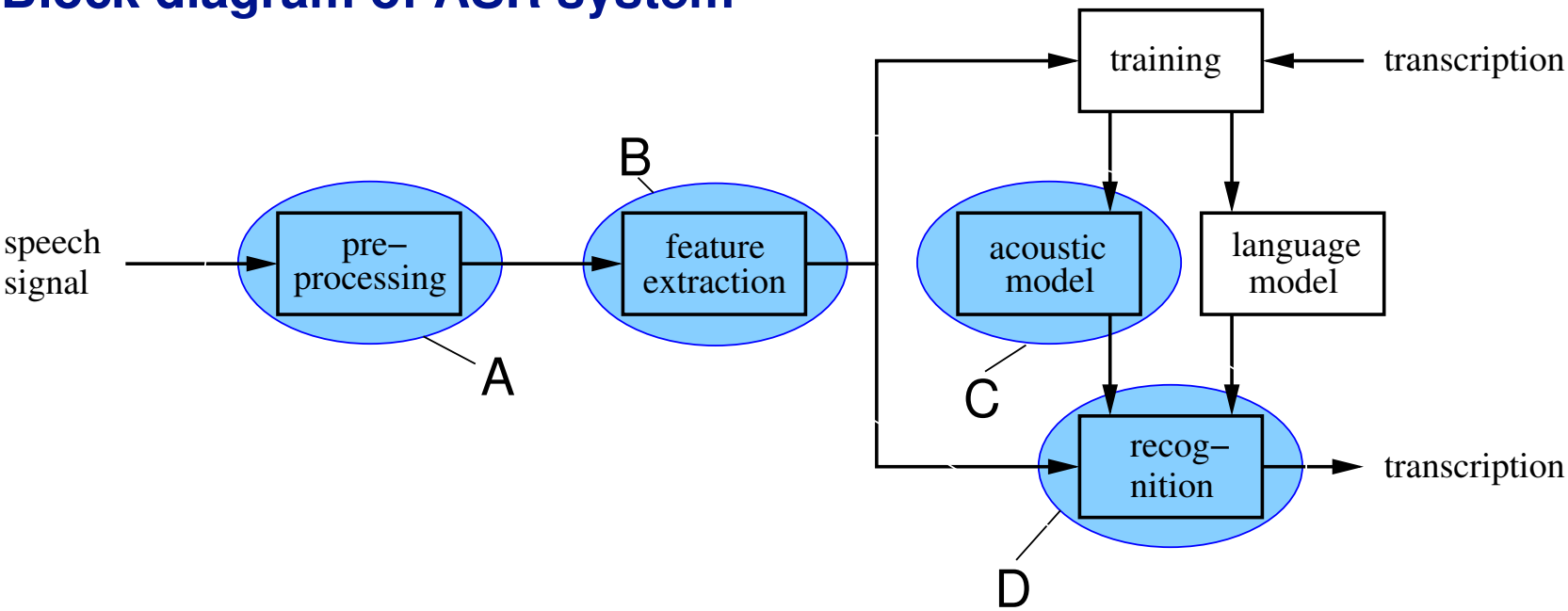
A) signal-based approaches

B) feature-based approaches

C) model-based approaches

Strategies for Reverberation-Robust ASR

Block diagram of ASR system



Strategies

A) signal-based approaches

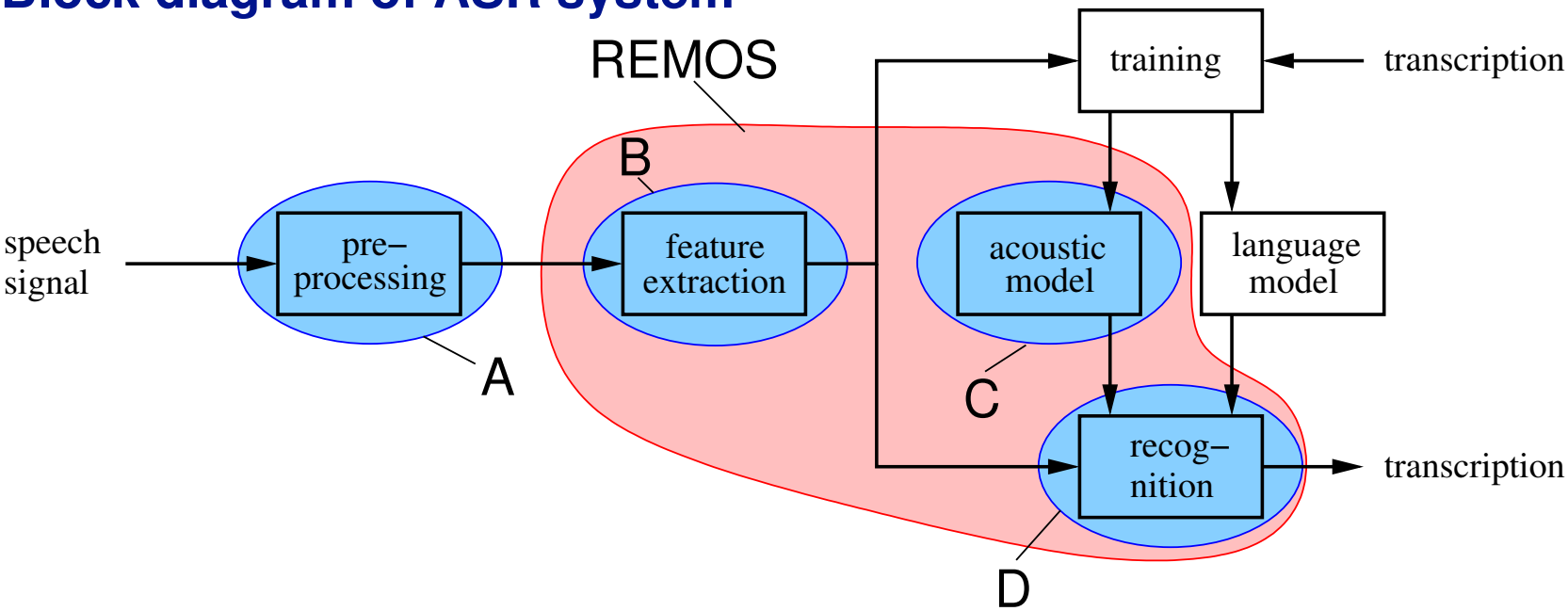
B) feature-based approaches

C) model-based approaches

D) decoder-based approaches

Strategies for Reverberation-Robust ASR

Block diagram of ASR system



Strategies

A) signal-based approaches

B) feature-based approaches

C) model-based approaches

D) decoder-based approaches

Part III: Robust ASR in Reverberant Environments

- Introduction
- Feature-based Approaches
- Model-based Approaches
- Decoder-based Approaches
- REMOS

Key Ideas of Feature-based Approaches

Three Different Approaches

- Feature compensation

  $\Rightarrow$ Example: Cepstral mean normalization (CMN)

Key Ideas of Feature-based Approaches

Three Different Approaches

- Feature compensation

  $\Rightarrow$ Example: Cepstral mean normalization (CMN)

- Features insensitive to reverberation

  $\Rightarrow$ Example: RASTA features

Key Ideas of Feature-based Approaches

Three Different Approaches

- Feature compensation
  $\Rightarrow$ Example: Cepstral mean normalization (CMN)
- Features insensitive to reverberation
  $\Rightarrow$ Example: RASTA features
- Features facilitating the capture of statistical properties
  $\Rightarrow$ Example: Dynamic features

Cepstral Mean Normalization [Atal 1974]

If impulse response much shorter than STFT analysis window

$$x_t \quad = \quad h_t * s_t$$

Cepstral Mean Normalization [Atal 1974]

If impulse response much shorter than STFT analysis window

$$x_t = h_t * s_t$$

$$|X_{n,k}^{\mathrm{STFT}}|^2 \approx |H_k^{\mathrm{STFT}}|^2 \, |S_{n,k}^{\mathrm{STFT}}|^2$$

Cepstral Mean Normalization [Atal 1974]

If impulse response much shorter than STFT analysis window

$$
\begin{aligned}
x_t &= h_t * s_t \\
|X_{n,k}^{\mathrm{STFT}}|^2 &\approx |H_k^{\mathrm{STFT}}|^2 \, |S_{n,k}^{\mathrm{STFT}}|^2 \\
x_{n,c}^{\mathrm{MFCC}} &\approx h_c^{\mathrm{MFCC}} + s_{n,c}^{\mathrm{MFCC}}
\end{aligned}
$$

Cepstral Mean Normalization [Atal 1974]

If impulse response much shorter than STFT analysis window

$$
\begin{aligned}
x_t &= h_t * s_t \\
|X_{n,k}^{\mathrm{STFT}}|^2 &\approx |H_k^{\mathrm{STFT}}|^2 \, |S_{n,k}^{\mathrm{STFT}}|^2 \\
x_{n,c}^{\mathrm{MFCC}} &\approx h_c^{\mathrm{MFCC}} + s_{n,c}^{\mathrm{MFCC}}
\end{aligned}
$$

Cepstral Mean Normalization

$$
x_{n,c}^{\mathrm{CMN}} = x_{n,c}^{\mathrm{MFCC}} - \bar{x}_c^{\mathrm{MFCC}}
$$

Cepstral Mean Normalization [Atal 1974]

If impulse response much shorter than STFT analysis window

$$
\begin{aligned}
x_t &= h_t * s_t \\
|X_{n,k}^{\mathrm{STFT}}|^2 &\approx |H_k^{\mathrm{STFT}}|^2 \, |S_{n,k}^{\mathrm{STFT}}|^2 \\
x_{n,c}^{\mathrm{MFCC}} &\approx h_c^{\mathrm{MFCC}} + s_{n,c}^{\mathrm{MFCC}}
\end{aligned}
$$

Cepstral Mean Normalization

$$
\begin{aligned}
x_{n,c}^{\mathrm{CMN}} &= x_{n,c}^{\mathrm{MFCC}} - \bar{x}_c^{\mathrm{MFCC}} \\
\bar{x}_c^{\mathrm{MFCC}} &= \frac{1}{N} \sum_{n=1}^{N} x_{n,c}^{\mathrm{MFCC}} \approx h_c^{\mathrm{MFCC}} + \bar{s}_c^{\mathrm{MFCC}}
\end{aligned}
$$

Cepstral Mean Normalization [Atal 1974]

If impulse response much shorter than STFT analysis window

$$
\begin{aligned}
x_t &= h_t * s_t \\
|X_{n,k}^{\mathrm{STFT}}|^2 &\approx |H_k^{\mathrm{STFT}}|^2 \, |S_{n,k}^{\mathrm{STFT}}|^2 \\
x_{n,c}^{\mathrm{MFCC}} &\approx h_c^{\mathrm{MFCC}} + s_{n,c}^{\mathrm{MFCC}}
\end{aligned}
$$

Cepstral Mean Normalization

$$
\begin{aligned}
x_{n,c}^{\mathrm{CMN}} &= x_{n,c}^{\mathrm{MFCC}} - \bar{x}_c^{\mathrm{MFCC}} \\
\bar{x}_c^{\mathrm{MFCC}} &= \frac{1}{N} \sum_{n=1}^{N} x_{n,c}^{\mathrm{MFCC}} \approx h_c^{\mathrm{MFCC}} + \bar{s}_c^{\mathrm{MFCC}} \\
x_{n,c}^{\mathrm{CMN}} &\approx h_c^{\mathrm{MFCC}} + s_{n,c}^{\mathrm{MFCC}} - (h_c^{\mathrm{MFCC}} + \bar{s}_c^{\mathrm{MFCC}}) = s_{n,c}^{\mathrm{MFCC}} - \bar{s}_c^{\mathrm{MFCC}} = s_{n,c}^{\mathrm{CMN}}
\end{aligned}
$$

Cepstral Mean Normalization [Atal 1974]

If impulse response much shorter than STFT analysis window

$$
\begin{aligned}
x_t &= h_t * s_t \\
|X_{n,k}^{\mathrm{STFT}}|^2 &\approx |H_k^{\mathrm{STFT}}|^2 \, |S_{n,k}^{\mathrm{STFT}}|^2 \\
x_{n,c}^{\mathrm{MFCC}} &\approx h_c^{\mathrm{MFCC}} + s_{n,c}^{\mathrm{MFCC}}
\end{aligned}
$$

Cepstral Mean Normalization

$$
x_{n,c}^{\mathrm{CMN}} = x_{n,c}^{\mathrm{MFCC}} - \bar{x}_c^{\mathrm{MFCC}}
$$

$$
\bar{x}_c^{\mathrm{MFCC}} = \frac{1}{N} \sum_{n=1}^{N} x_{n,c}^{\mathrm{MFCC}} \approx h_c^{\mathrm{MFCC}} + \bar{s}_c^{\mathrm{MFCC}}
$$

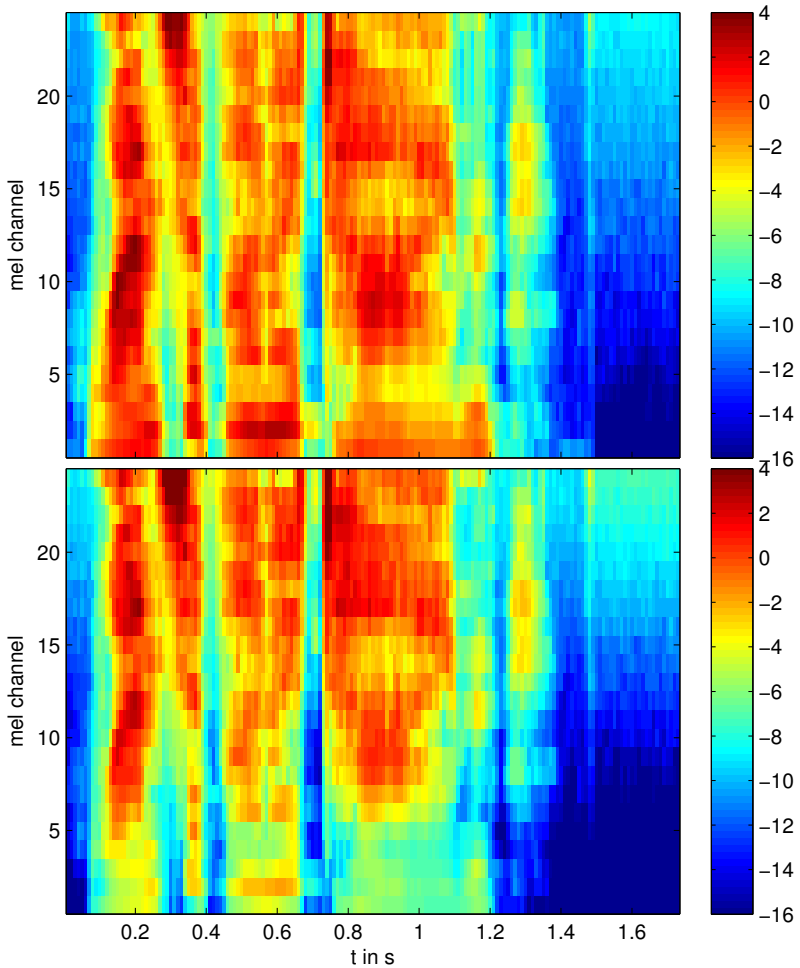
$$
x_{n,c}^{\mathrm{CMN}} \approx h_c^{\mathrm{MFCC}} + s_{n,c}^{\mathrm{MFCC}} - (h_c^{\mathrm{MFCC}} + \bar{s}_c^{\mathrm{MFCC}}) = s_{n,c}^{\mathrm{MFCC}} - \bar{s}_c^{\mathrm{MFCC}} = s_{n,c}^{\mathrm{CMN}}
$$

$\Rightarrow$ convolution compensated

CMN - Illustration 1st-order Highpass Filter

Clean vs. Highpass Filtered Logmel Features

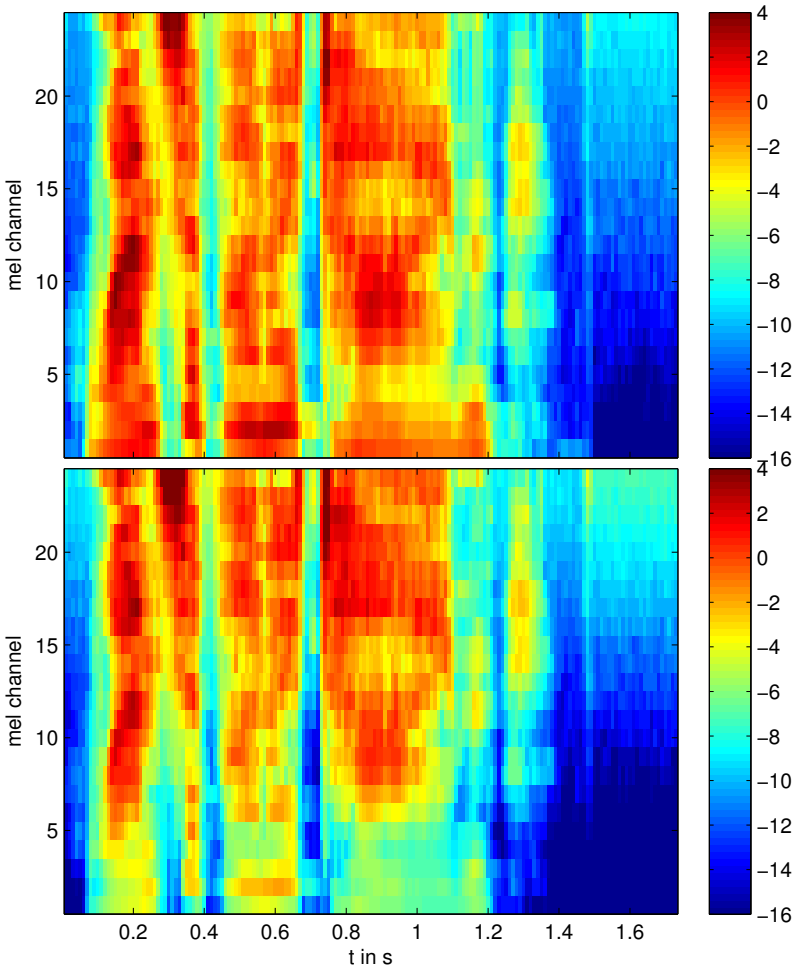
No CMN



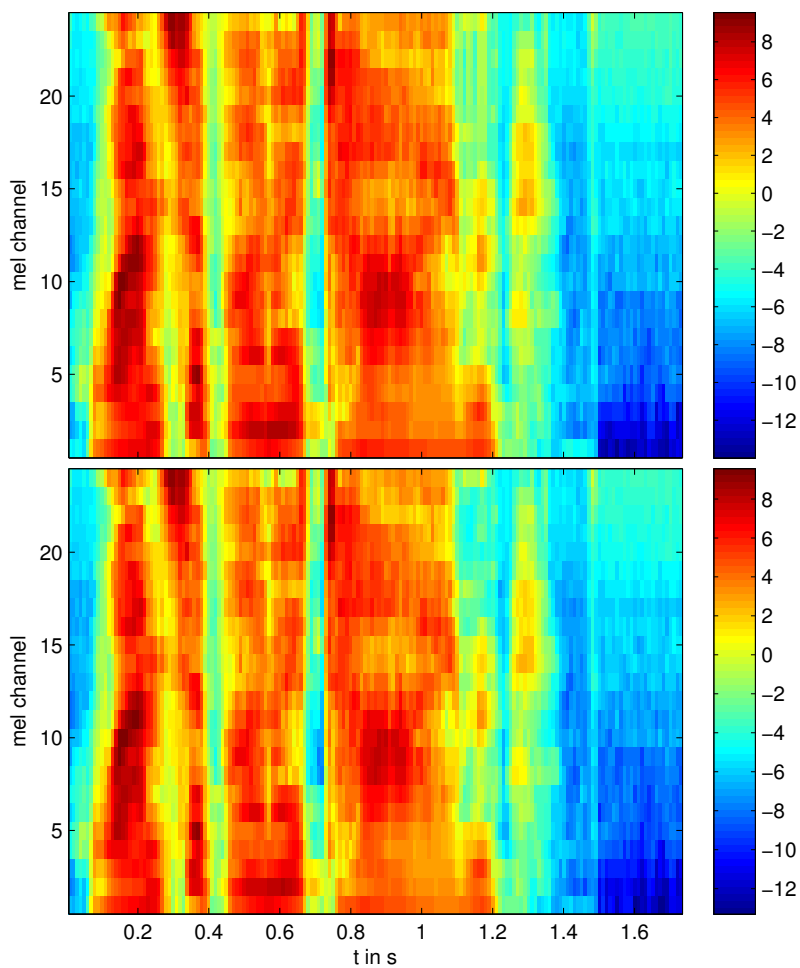
CMN - Illustration 1st-order Highpass Filter

Clean vs. Highpass Filtered Logmel Features

No CMN



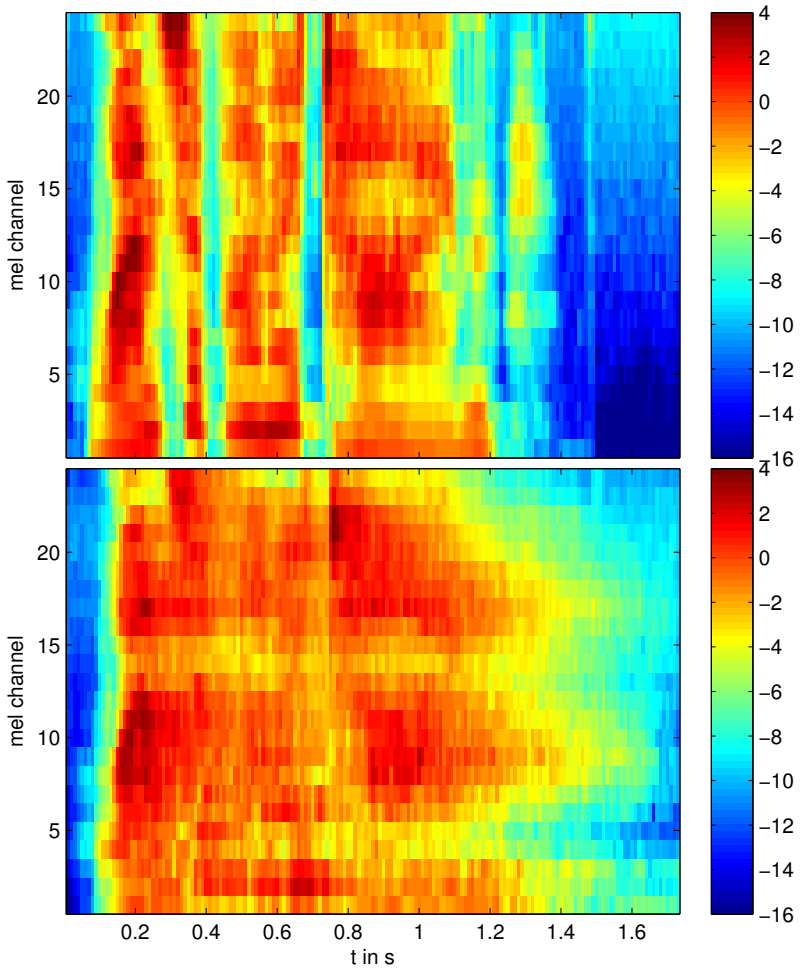
With CMN



CMN - Illustration Reverberation

Clean vs. Reverberant ($T_{60} = 900\,\mathrm{ms}$) Logmel Features

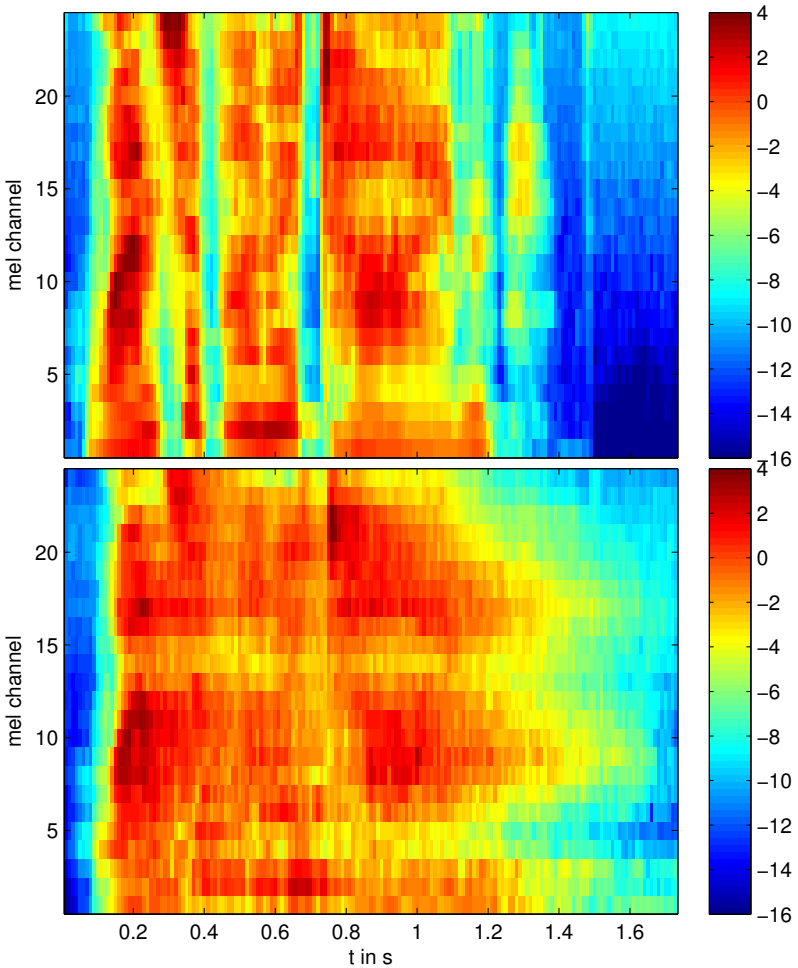
No CMN



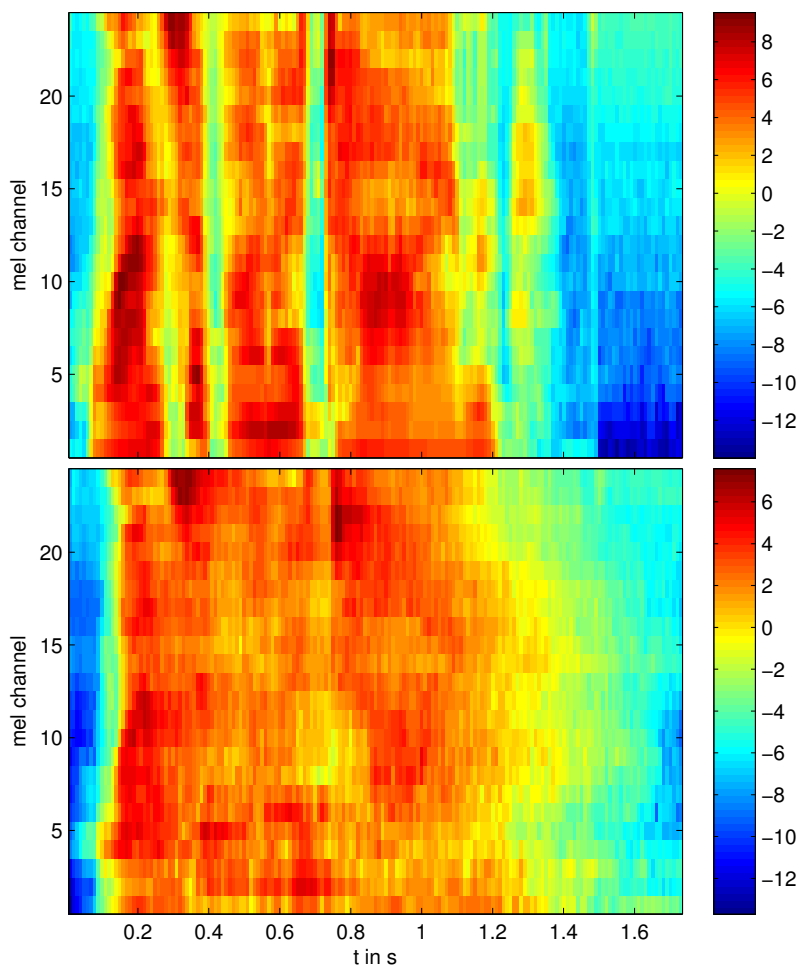
CMN - Illustration Reverberation

Clean vs. Reverberant ($T_{60} = 900\,\mathrm{ms}$) Logmel Features

No CMN



With CMN



CMN - Discussion

Approach

- Apply CMN to both training and test data

CMN - Discussion

Approach

- Apply CMN to both training and test data

$\Rightarrow$ Short impulse responses can be compensated

$+$ Good for compensating different microphone characteristics or different telephone channels

$+$ Good for compensating coloration due to early reflections

CMN - Discussion

Approach

- ▸ Apply CMN to both training and test data
- $\Rightarrow$ Short impulse responses can be compensated
- $+$ Good for compensating different microphone characteristics or different telephone channels
- $+$ Good for compensating coloration due to early reflections
- $-$ But: not suitable for compensating late reverberation

CMN - Discussion

Approach

- Apply CMN to both training and test data

$\Rightarrow$ Short impulse responses can be compensated

$+$ Good for compensating different microphone characteristics or different telephone channels

$+$ Good for compensating coloration due to early reflections

$-$ But: not suitable for compensating late reverberation

Further considerations

- Reliable only if utterance is long enough ($>$4 s [Droppo 2008])
- Extensions necessary for different speech activity rates of training and test data [Droppo 2008]

RASTA (RelAtive SpecTrA) Features [Hermansky 1994]

Background

- Speed of spectral changes of speech:
  $\Rightarrow$ limited by movements of articulators in vocal tract

RASTA (RelAtive SpecTrA) Features [Hermansky 1994]

Background

- Speed of spectral changes of speech:
  $\Rightarrow$ limited by movements of articulators in vocal tract
- Many non-speech effects:
  $\Rightarrow$ characterized by short time-invariant impulse responses
  Examples: microphone characteristics, telephone channels

RASTA (RelAtive SpecTrA) Features [Hermansky 1994]

Background

- Speed of spectral changes of speech:
  $\Rightarrow$ limited by movements of articulators in vocal tract
- Many non-speech effects:
  $\Rightarrow$ characterized by short time-invariant impulse responses
  Examples: microphone characteristics, telephone channels
- Analysis artifacts:
  $\Rightarrow$ very fast spectral changes

RASTA (RelAtive SpecTrA) Features [Hermansky 1994]

Background

- Speed of spectral changes of speech:
  $\Rightarrow$ limited by movements of articulators in vocal tract
- Many non-speech effects:
  $\Rightarrow$ characterized by short time-invariant impulse responses
  Examples: microphone characteristics, telephone channels
- Analysis artifacts:
  $\Rightarrow$ very fast spectral changes

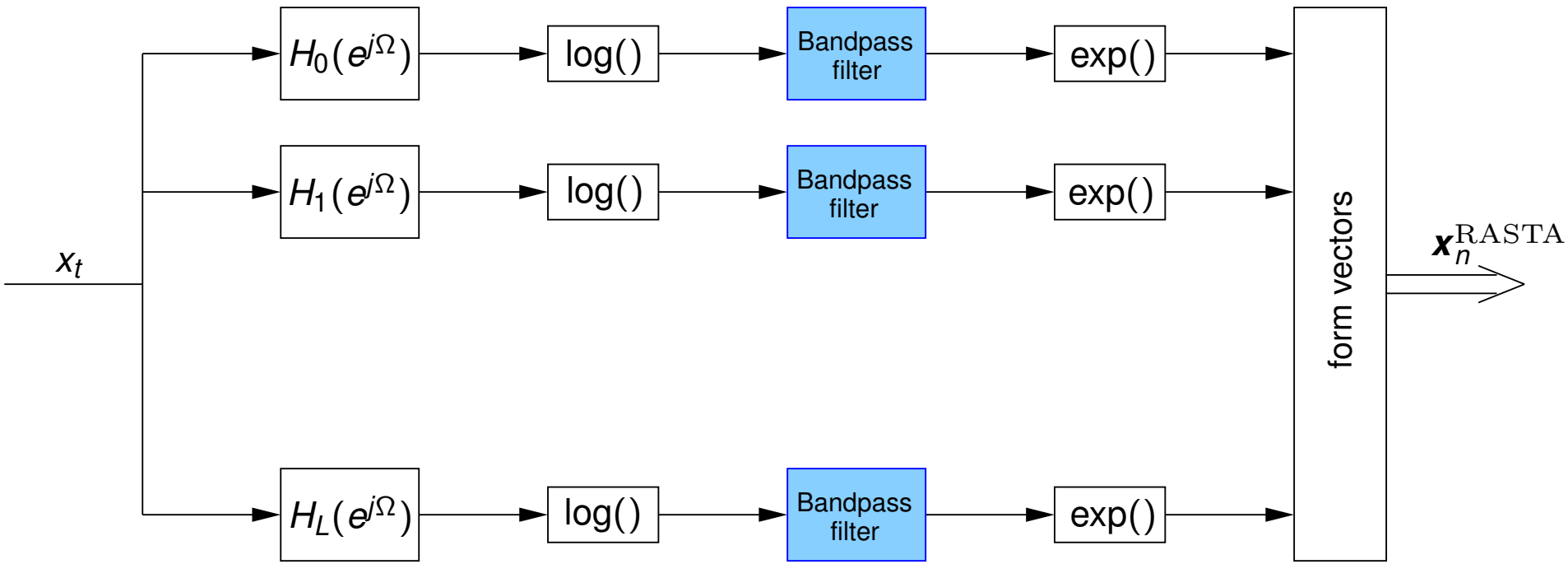
Idea

- Remove very slow and fast spectral changes from features:
  $\Rightarrow$ bandpass filtering in each channel

\+ Insensitivity to slow and fast spectral changes

RASTA Features: Block Diagram

Calculation of RASTA Features



RASTA Features: Discussion

RASTA Features

- Effective for short time-invariant impulse responses (like CMN)
- \+ Good for compensating different microphone characteristics or different telephone channels
- \+ Good for compensating coloration due to early reflections

RASTA Features: Discussion

RASTA Features

- ▸ Effective for short time-invariant impulse responses (like CMN)
- \+ Good for compensating different microphone characteristics or different telephone channels
- \+ Good for compensating coloration due to early reflections
- ▸ Reverberation described by long RIRs
- – Therefore: not suitable for compensating late reverberation

Dynamic Features [Furui 1986]

Idea

- Temporal changes of short-time spectra:
  $\Rightarrow$ important for discriminating phonemes
- First and second derivate of static features (Δ and $\Delta\Delta$ features):
  $\Rightarrow$ capture these changes

Dynamic Features [Furui 1986]

Idea

- Temporal changes of short-time spectra:
  $\Rightarrow$ important for discriminating phonemes
- First and second derivate of static features (Δ and $\Delta\Delta$ features):
  $\Rightarrow$ capture these changes

Δ Feature Calculation

$$\Delta \boldsymbol{s}_n = \boldsymbol{s}_{n+\kappa} - \boldsymbol{s}_{n-\kappa}$$

typical: $\kappa \in \{1, 2\}$

Dynamic Features [Furui 1986]

Idea

- Temporal changes of short-time spectra:
  $\Rightarrow$ important for discriminating phonemes
- First and second derivate of static features (Δ and $\Delta\Delta$ features):
  $\Rightarrow$ capture these changes

Δ Feature Calculation

$$\Delta \boldsymbol{s}_n = \boldsymbol{s}_{n+\kappa} - \boldsymbol{s}_{n-\kappa}$$

$$\Delta \boldsymbol{s}_n = \frac{\sum_{\kappa=1}^{N_\Delta} \kappa \cdot \left(\boldsymbol{s}_{n+\kappa} - \boldsymbol{s}_{n-\kappa} \right)}{2 \cdot \sum_{\kappa=1}^{N_\Delta} \kappa^2}$$

typical: $\kappa \in \{1, 2\}$ or $N_\Delta \in \{2, 3, 4\}$

Dynamic Features [Furui 1986]

Idea

- Temporal changes of short-time spectra:
  $\Rightarrow$ important for discriminating phonemes
- First and second derivate of static features (Δ and $\Delta\Delta$ features):
  $\Rightarrow$ capture these changes

Δ Feature Calculation

$$\Delta \boldsymbol{s}_n = \boldsymbol{s}_{n+\kappa} - \boldsymbol{s}_{n-\kappa}$$

$$\Delta \boldsymbol{s}_n = \frac{\sum_{\kappa=1}^{N_\Delta} \kappa \cdot \left(\boldsymbol{s}_{n+\kappa} - \boldsymbol{s}_{n-\kappa} \right)}{2 \cdot \sum_{\kappa=1}^{N_\Delta} \kappa^2}$$

typical: $\kappa \in \{1, 2\}$ or $N_\Delta \in \{2, 3, 4\}$

$\Delta\Delta$ features: calculated in a similar way from Δ features

Why are Dynamic Features interesting for Reverberant ASR?

Reverberant Speech

- Long-term relations between feature vectors
- Cannot be captured by HMMs

Why are Dynamic Features interesting for Reverberant ASR?

Reverberant Speech

- Long-term relations between feature vectors
- Cannot be captured by HMMs
  $\Rightarrow$ Mitigation by feature vectors with long temporal reach

Why are Dynamic Features interesting for Reverberant ASR?

Reverberant Speech

- Long-term relations between feature vectors
- Cannot be captured by HMMs
  $\Rightarrow$ Mitigation by feature vectors with long temporal reach

Temporal reach of features

- Static features: typically 10 ms – 40 ms
- Δ features: typically 20 ms – 120 ms
- $\Delta\Delta$ features: typically 30 ms – 200 ms

Why are Dynamic Features interesting for Reverberant ASR?

Reverberant Speech

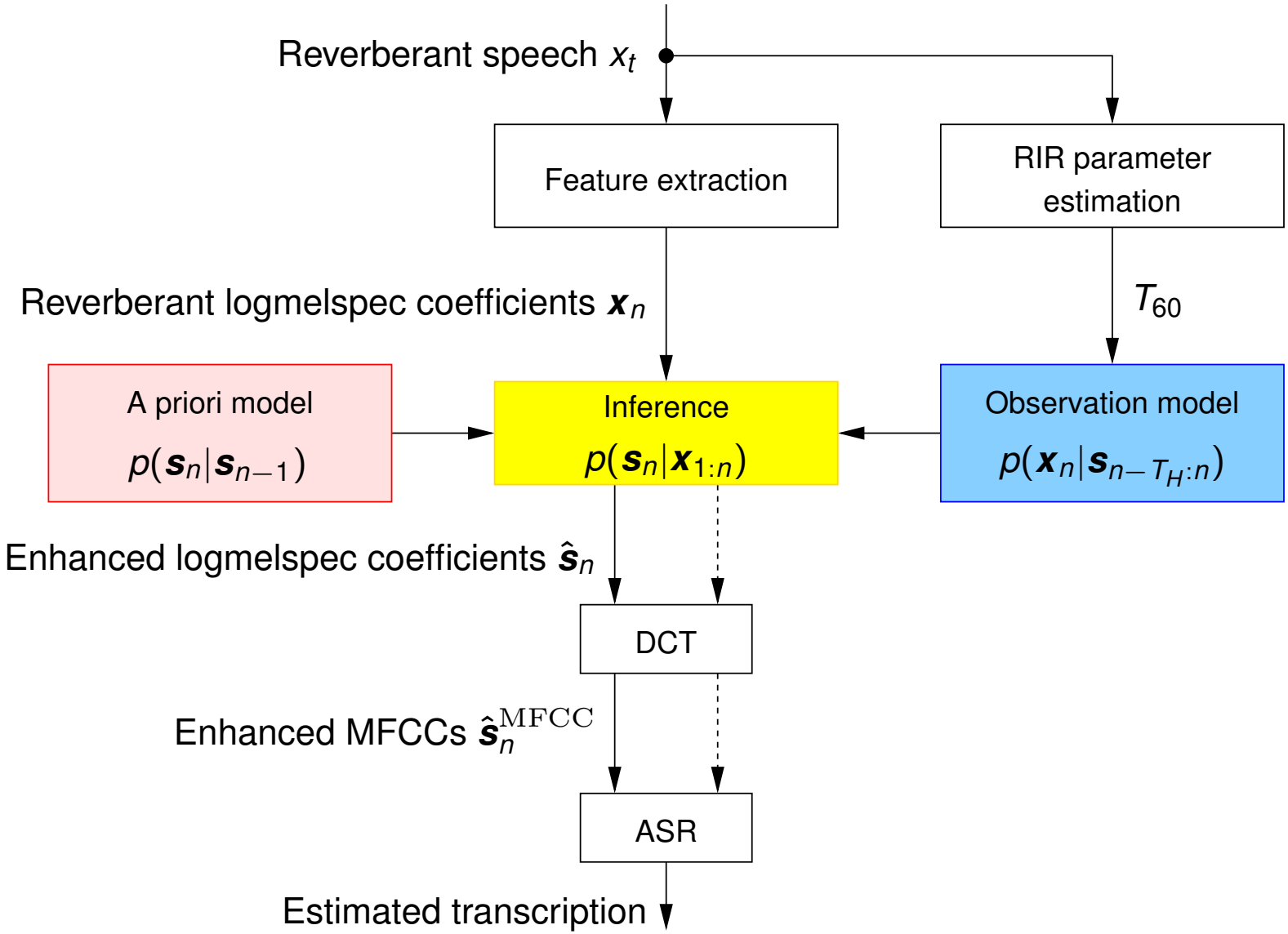
- Long-term relations between feature vectors
- Cannot be captured by HMMs
  $\Rightarrow$ Mitigation by feature vectors with long temporal reach

Temporal reach of features

- Static features: typically 10 ms – 40 ms
- Δ features: typically 20 ms – 120 ms
- $\Delta\Delta$ features: typically 30 ms – 200 ms

Dynamic features can partly capture long-term relations

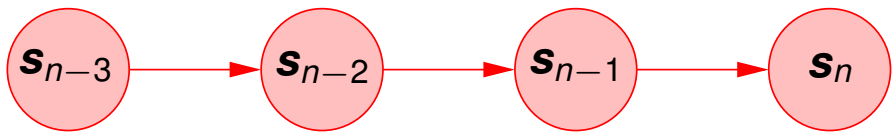
Model-based Feature Enhancement [Krueger 2010]



Model-based Feature Enhancement [Krueger 2010]

A Priori Model: Clean Speech Model

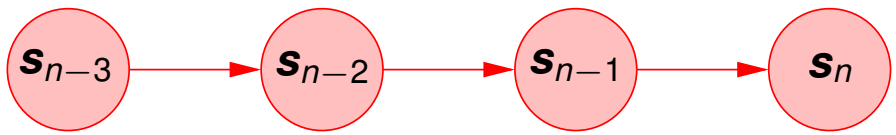
Linear dynamic model



Model-based Feature Enhancement [Krueger 2010]

A Priori Model: Clean Speech Model

Linear dynamic model

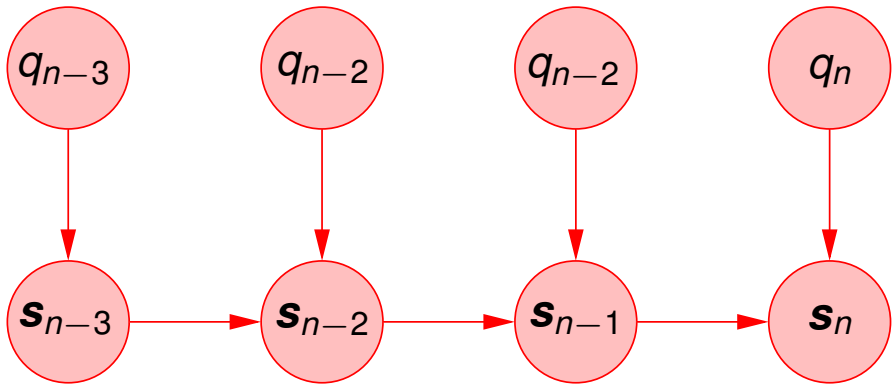


$$\boldsymbol{s}_n = \boldsymbol{A}\boldsymbol{s}_{n-1} + \boldsymbol{b} + \boldsymbol{u}_n$$

Model-based Feature Enhancement [Krueger 2010]

A Priori Model: Clean Speech Model

Switching linear dynamic model

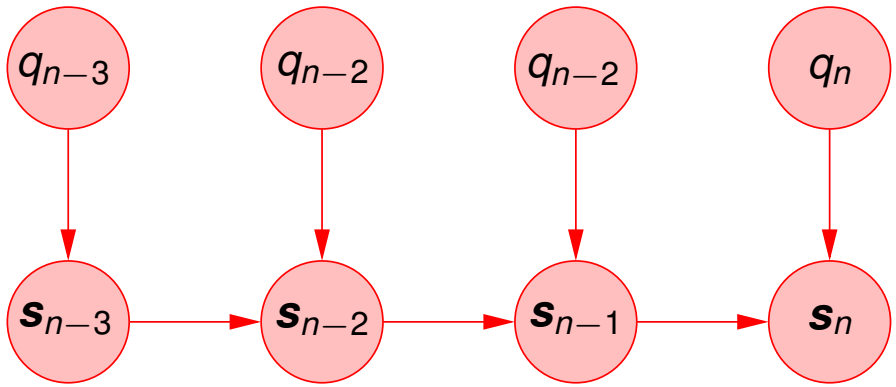


$$\boldsymbol{s}_n = \boldsymbol{A}(q_n)\boldsymbol{s}_{n-1} + \boldsymbol{b}(q_n) + \boldsymbol{u}_n$$

Model-based Feature Enhancement [Krueger 2010]

A Priori Model: Clean Speech Model

Switching linear dynamic model



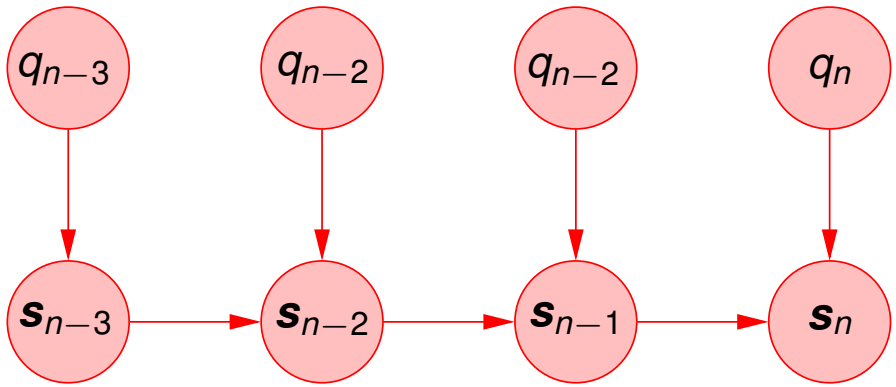
$$\boldsymbol{s}_n = \boldsymbol{A}(q_n)\boldsymbol{s}_{n-1} + \boldsymbol{b}(q_n) + \boldsymbol{u}_n$$

$$p(\boldsymbol{s}_n|\boldsymbol{s}_{n-1}, q_n) = \mathcal{N}(\boldsymbol{s}_n; \boldsymbol{A}(q_n)\boldsymbol{s}_{n-1} + \boldsymbol{b}(q_n), \Sigma_{\boldsymbol{u}}(q_n))$$

Model-based Feature Enhancement [Krueger 2010]

A Priori Model: Clean Speech Model

Switching linear dynamic model



$$\boldsymbol{s}_n = \boldsymbol{A}(q_n)\boldsymbol{s}_{n-1} + \boldsymbol{b}(q_n) + \boldsymbol{u}_n$$

$$p(\boldsymbol{s}_n|\boldsymbol{s}_{n-1}, q_n) = \mathcal{N}(\boldsymbol{s}_n; \boldsymbol{A}(q_n)\boldsymbol{s}_{n-1} + \boldsymbol{b}(q_n), \boldsymbol{\Sigma}_{\boldsymbol{u}}(q_n))$$

Model for non-stationary feature vector sequences of clean speech

Model-based Feature Enhancement [Krueger 2010]

Observation Model: Reverberation Model

Model-based Feature Enhancement [Krueger 2010]

Observation Model: Reverberation Model

based on melspec convolution

$$\boldsymbol{x}_n = \log\left(\sum_{\tau=0}^{T_H} \exp(\boldsymbol{h}_\tau + \boldsymbol{s}_{n-\tau})\right) + \boldsymbol{v}_n$$

$\boldsymbol{v}_n$: captures approximation error

$\boldsymbol{h}_{0:T_H}$: based on strictly exponentially decaying RIR model
$\Rightarrow$ Only T_{60} needs to be estimated

Model-based Feature Enhancement [Krueger 2010]

Observation Model: Reverberation Model

based on melspec convolution

$$\begin{aligned} \boldsymbol{x}_n &= \log\left(\sum_{\tau=0}^{T_H} \exp(\boldsymbol{h}_\tau + \boldsymbol{s}_{n-\tau})\right) + \boldsymbol{v}_n \\ &= f\left(\boldsymbol{s}_{n-T_H:n}, \boldsymbol{h}_{0:T_H}\right) + \boldsymbol{v}_n \end{aligned}$$

$\boldsymbol{v}_n$: captures approximation error

$\boldsymbol{h}_{0:T_H}$: based on strictly exponentially decaying RIR model
$\Rightarrow$ Only T_{60} needs to be estimated

Model-based Feature Enhancement [Krueger 2010]

Observation Model: Reverberation Model

based on melspec convolution

$$\begin{aligned} \boldsymbol{x}_n &= \log\left(\sum_{\tau=0}^{T_H} \exp(\boldsymbol{h}_\tau + \boldsymbol{s}_{n-\tau})\right) + \boldsymbol{v}_n \\ &= f\left(\boldsymbol{s}_{n-T_H:n}, \boldsymbol{h}_{0:T_H}\right) + \boldsymbol{v}_n \\ p(\boldsymbol{v}_n) &= \mathcal{N}\left(\boldsymbol{v}_n; \boldsymbol{\mu}_{\boldsymbol{v}}, \boldsymbol{\Sigma}_{\boldsymbol{v}}\right) \end{aligned}$$

$\boldsymbol{v}_n$: captures approximation error

$\boldsymbol{h}_{0:T_H}$: based on strictly exponentially decaying RIR model
$\Rightarrow$ Only T_{60} needs to be estimated

Model-based Feature Enhancement [Krueger 2010]

Observation Model: Reverberation Model

based on melspec convolution

$$
\begin{aligned}
\boldsymbol{x}_n &= \log\left(\sum_{\tau=0}^{T_H} \exp(\boldsymbol{h}_\tau + \boldsymbol{s}_{n-\tau})\right) + \boldsymbol{v}_n \\
&= f\left(\boldsymbol{s}_{n-T_H:n}, \boldsymbol{h}_{0:T_H}\right) + \boldsymbol{v}_n \\
p(\boldsymbol{v}_n) &= \mathcal{N}\left(\boldsymbol{v}_n; \boldsymbol{\mu}_{\boldsymbol{v}}, \Sigma_{\boldsymbol{v}}\right) \\
p(\boldsymbol{x}_n | \boldsymbol{s}_{n-T_H:n}) &= \mathcal{N}\left(\boldsymbol{x}_n; f\left(\boldsymbol{s}_{n-T_H:n}, \boldsymbol{h}_{0:T_H}\right) + \boldsymbol{\mu}_{\boldsymbol{v}}, \Sigma_{\boldsymbol{v}}\right)
\end{aligned}
$$

$\boldsymbol{v}_n$: captures approximation error

$\boldsymbol{h}_{0:T_H}$: based on strictly exponentially decaying RIR model
$\Rightarrow$ Only T_{60} needs to be estimated

Model-based Feature Enhancement [Krueger 2010]

Bayesian Inference

Model-based Feature Enhancement [Krueger 2010]

Bayesian Inference

MMSE estimate

$$\hat{\boldsymbol{s}}_n = \mathrm{E}\{\boldsymbol{s}_n | \boldsymbol{x}_{1:n}\}$$

Model-based Feature Enhancement [Krueger 2010]

Bayesian Inference

MMSE estimate

$$\hat{\boldsymbol{s}}_n = \mathrm{E}\{\boldsymbol{s}_n|\boldsymbol{x}_{1:n}\}$$

$$p(\boldsymbol{s}_n|\boldsymbol{x}_{1:n}) = \frac{p(\boldsymbol{x}_n|\boldsymbol{s}_n,\boldsymbol{x}_{1:n-1})\ p(\boldsymbol{s}_n|\boldsymbol{x}_{1:n-1})}{\int p(\boldsymbol{x}_n|\boldsymbol{s}_n,\boldsymbol{x}_{1:n-1})\ p(\boldsymbol{s}_n|\boldsymbol{x}_{1:n-1})\mathrm{d}\boldsymbol{s}_n}$$

Model-based Feature Enhancement [Krueger 2010]

Bayesian Inference

MMSE estimate

$$\hat{\boldsymbol{s}}_n = \mathrm{E}\{\boldsymbol{s}_n|\boldsymbol{x}_{1:n}\}$$

$$p(\boldsymbol{s}_n|\boldsymbol{x}_{1:n}) = \frac{p(\boldsymbol{x}_n|\boldsymbol{s}_n,\boldsymbol{x}_{1:n-1})\ p(\boldsymbol{s}_n|\boldsymbol{x}_{1:n-1})}{\int p(\boldsymbol{x}_n|\boldsymbol{s}_n,\boldsymbol{x}_{1:n-1})\ p(\boldsymbol{s}_n|\boldsymbol{x}_{1:n-1})\mathrm{d}\boldsymbol{s}_n}$$

$$\approx \frac{p(\boldsymbol{x}_n|\boldsymbol{s}_{n-T_H:n})\ \sum_{i=1}^{M} p(\boldsymbol{s}_n|\boldsymbol{s}_{n-1},q_n=i)\ p(q_n=i)}{\int p(\boldsymbol{x}_n|\boldsymbol{s}_n,\boldsymbol{x}_{1:n-1})\ \sum_{i=1}^{M} p(\boldsymbol{s}_n|\boldsymbol{s}_{n-1},q_n=i)\ p(q_n=i)\ \mathrm{d}\boldsymbol{s}_n}$$

Model-based Feature Enhancement [Krueger 2010]

Bayesian Inference

MMSE estimate

$$\hat{\boldsymbol{s}}_n = \mathrm{E}\{\boldsymbol{s}_n|\boldsymbol{x}_{1:n}\}$$

$$p(\boldsymbol{s}_n|\boldsymbol{x}_{1:n}) = \frac{p(\boldsymbol{x}_n|\boldsymbol{s}_n,\boldsymbol{x}_{1:n-1})\ p(\boldsymbol{s}_n|\boldsymbol{x}_{1:n-1})}{\int p(\boldsymbol{x}_n|\boldsymbol{s}_n,\boldsymbol{x}_{1:n-1})\ p(\boldsymbol{s}_n|\boldsymbol{x}_{1:n-1})\mathrm{d}\boldsymbol{s}_n}$$

$$\approx \frac{p(\boldsymbol{x}_n|\boldsymbol{s}_{n-T_H:n})\ \sum_{i=1}^{M} p(\boldsymbol{s}_n|\boldsymbol{s}_{n-1},q_n=i)\ p(q_n=i)}{\int p(\boldsymbol{x}_n|\boldsymbol{s}_n,\boldsymbol{x}_{1:n-1})\ \sum_{i=1}^{M} p(\boldsymbol{s}_n|\boldsymbol{s}_{n-1},q_n=i)\ p(q_n=i)\ \mathrm{d}\boldsymbol{s}_n}$$

$\Rightarrow$ Inference performed by bank of iterated extended Kalman filters

Model-based Feature Enhancement [Krueger 2010]

Discussion

- Approach tailored to reverberant feature vector sequences
- long-term relations explicitely captured by observation model

Model-based Feature Enhancement [Krueger 2010]

Discussion

- Approach tailored to reverberant feature vector sequences
- long-term relations explicitely captured by observation model
- \+ Promising results reported on AURORA 5 task (connected digits)
- \+ Moderate computational complexity
- \+ Latency of only a few frames

Model-based Feature Enhancement [Krueger 2010]

Discussion

- Approach tailored to reverberant feature vector sequences
- long-term relations explicitely captured by observation model
- \+ Promising results reported on AURORA 5 task (connected digits)
- \+ Moderate computational complexity
- \+ Latency of only a few frames

Suitable for online recognition in reverberant environments

Further Feature-based Approaches

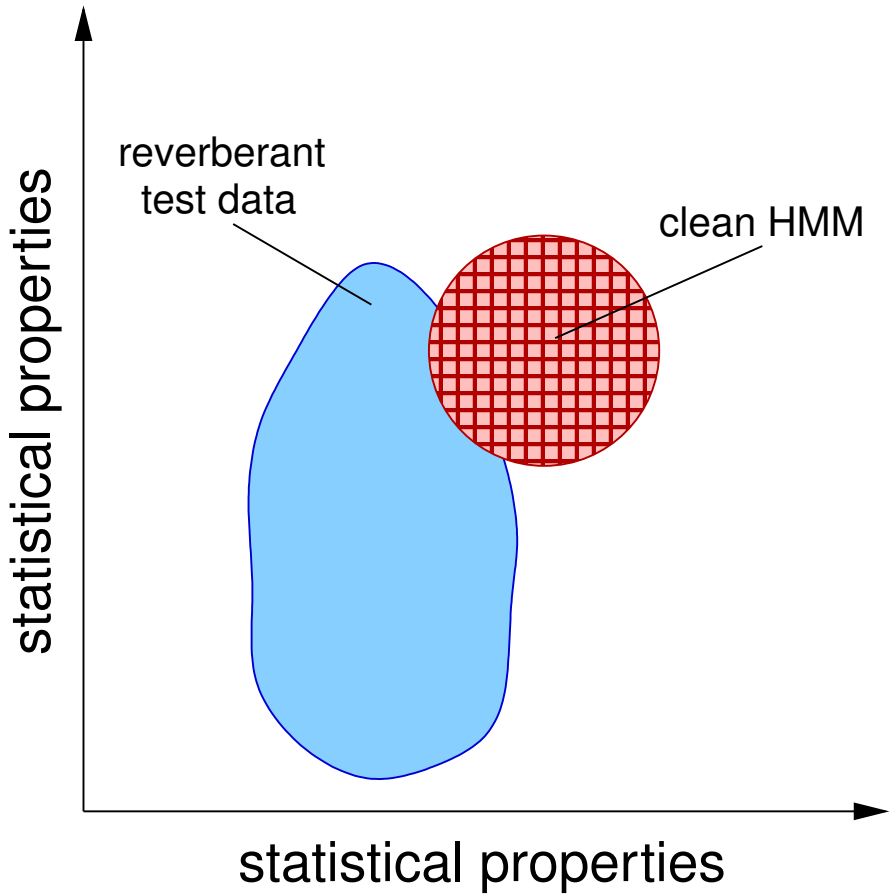
[Petrick 2008]	Harmonicity-based feature analysis
[Thomas 2008]	Frequency-domain linear prediction
[Wölfel 2009]	Particle filter-based feature enhancement
[Kumar 2010]	Cepstral inverse filtering

Part III: Robust ASR in Reverberant Environments

- Introduction
- Feature-based Approaches
- Model-based Approaches
- Decoder-based Approaches
- REMOS

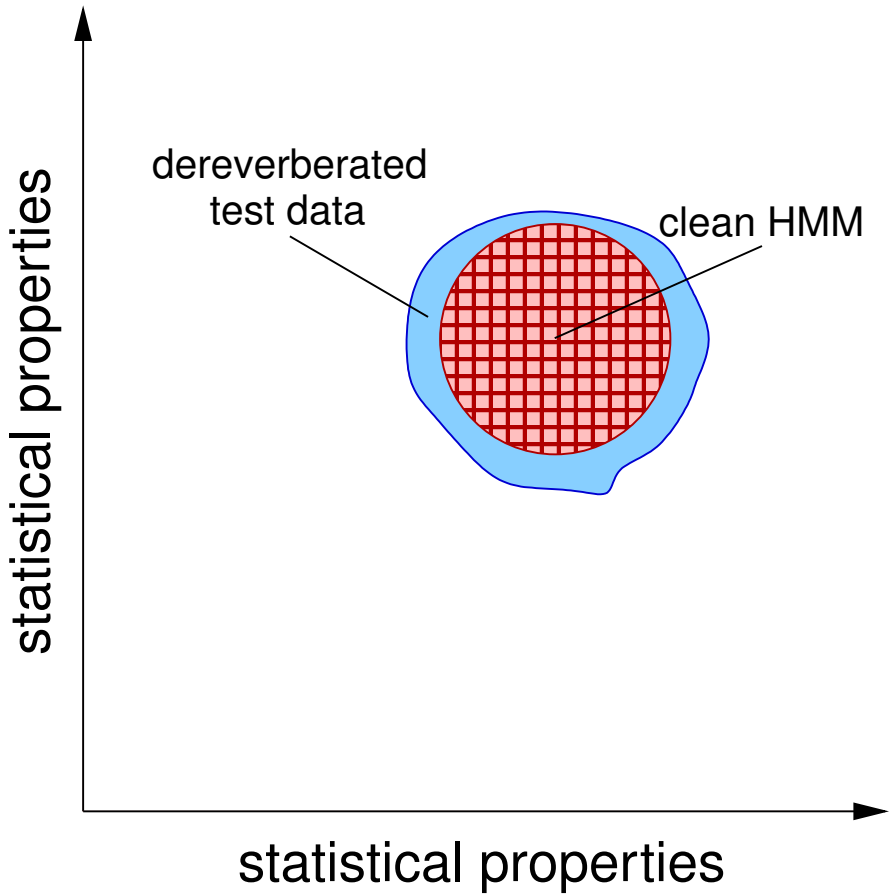
Key Idea of Model-based Approaches

- Mismatch between clean HMM and reverberant data



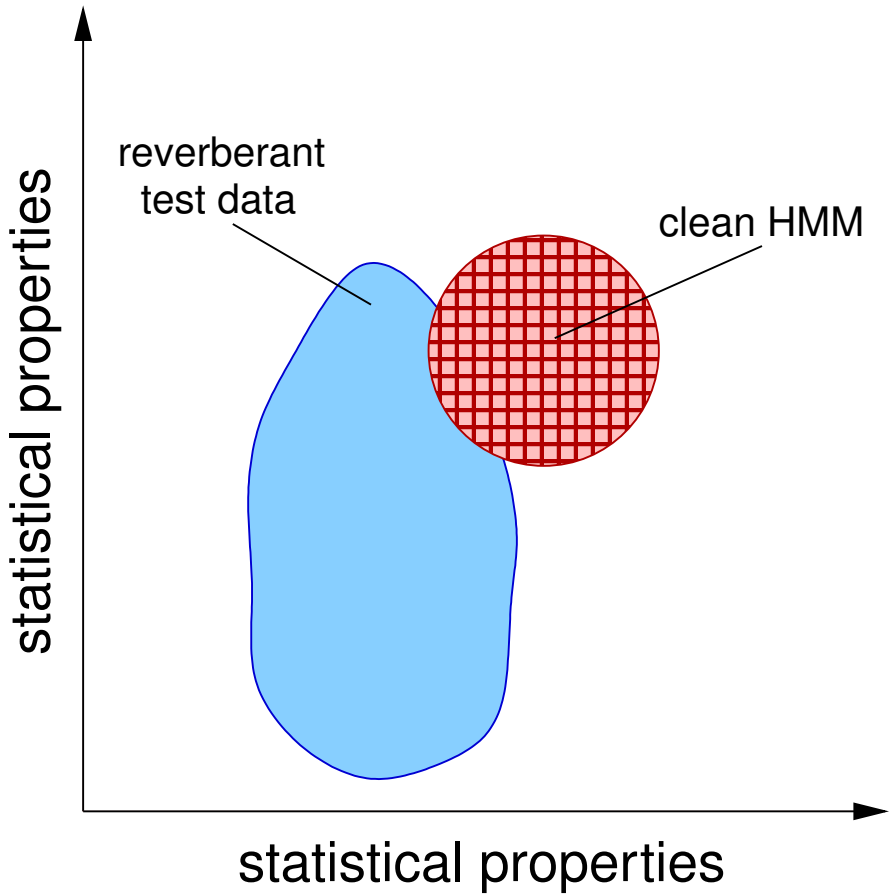
Key Idea of Model-based Approaches

- Feature-based: “dereverberate” data



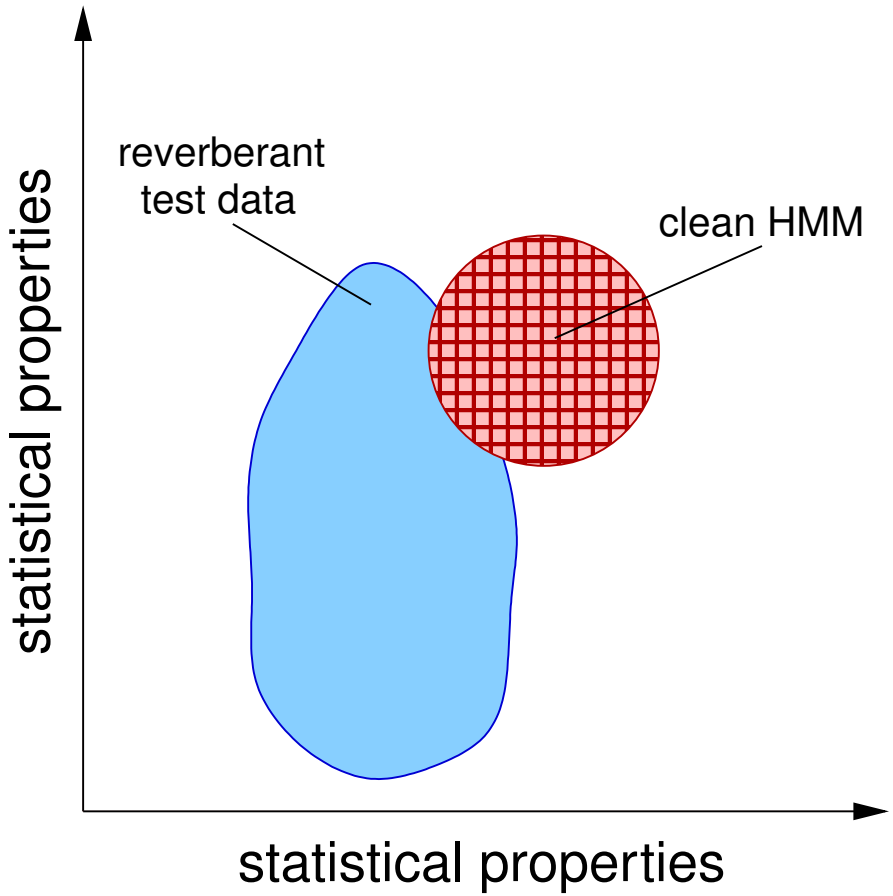
Key Idea of Model-based Approaches

- Model-based: “reverberate” acoustic model



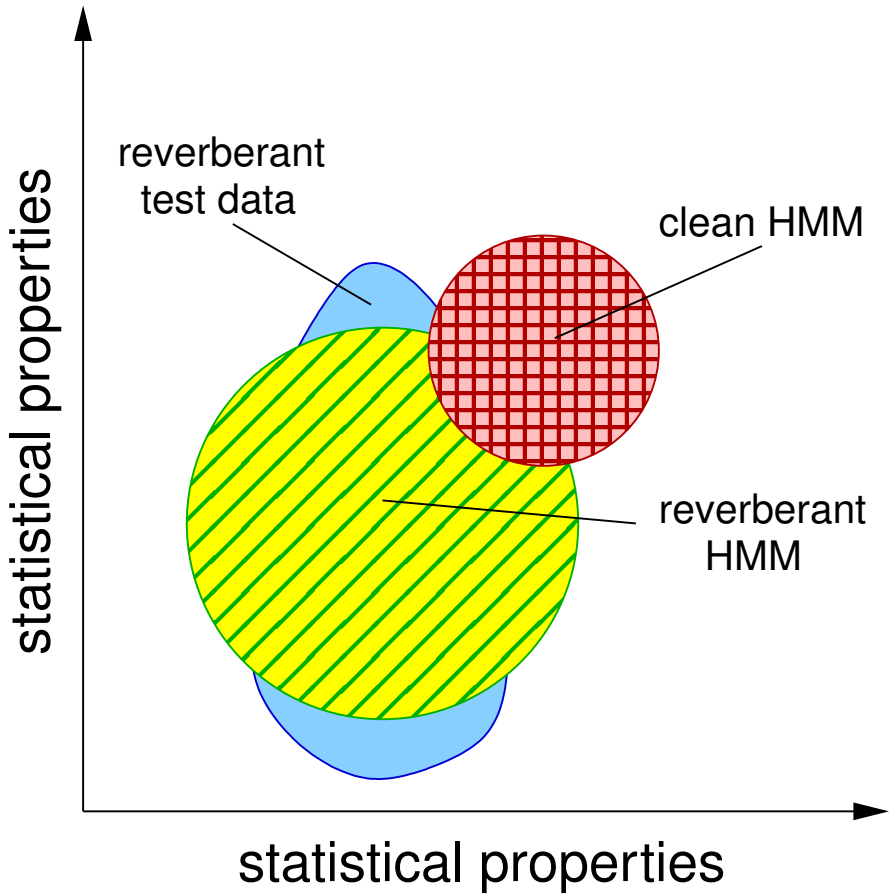
Key Idea of Model-based Approaches

- Model-based: “reverberate” acoustic model
- Adjust acoustic model to statistical properties of reverberant data



Key Idea of Model-based Approaches

- Model-based: “reverberate” acoustic model
- Adjust acoustic model to statistical properties of reverberant data



Training with Reverberant Data

Matched Training

- Record training data in target environment

Training with Reverberant Data

Matched Training

- Record training data in target environment
  - \+ Training data perfectly capture statistical properties
  - – Extremely high effort

Training with Reverberant Data

Matched Training

- Record training data in target environment
  - \+ Training data perfectly capture statistical properties
  - – Extremely high effort
- Generate training data by convolution with RIR [Giuliani 1999, Stahl 2001, Matassoni 2002]

Training with Reverberant Data

Matched Training

- Record training data in target environment
  - \+ Training data perfectly capture statistical properties
  - − Extremely high effort
- Generate training data by convolution with RIR [Giuliani 1999, Stahl 2001, Matassoni 2002]
  - \+ Significantly reduced effort
  - \+ Only slight degradation in recognition performance [Stahl 2001]

Training with Reverberant Data

Matched Training

- Record training data in target environment
  - \+ Training data perfectly capture statistical properties
  - − Extremely high effort
- Generate training data by convolution with RIR [Giuliani 1999, Stahl 2001, Matassoni 2002]
  - \+ Significantly reduced effort
  - \+ Only slight degradation in recognition performance [Stahl 2001]

Multi-Style Training

- Use training data from many different rooms

Training with Reverberant Data

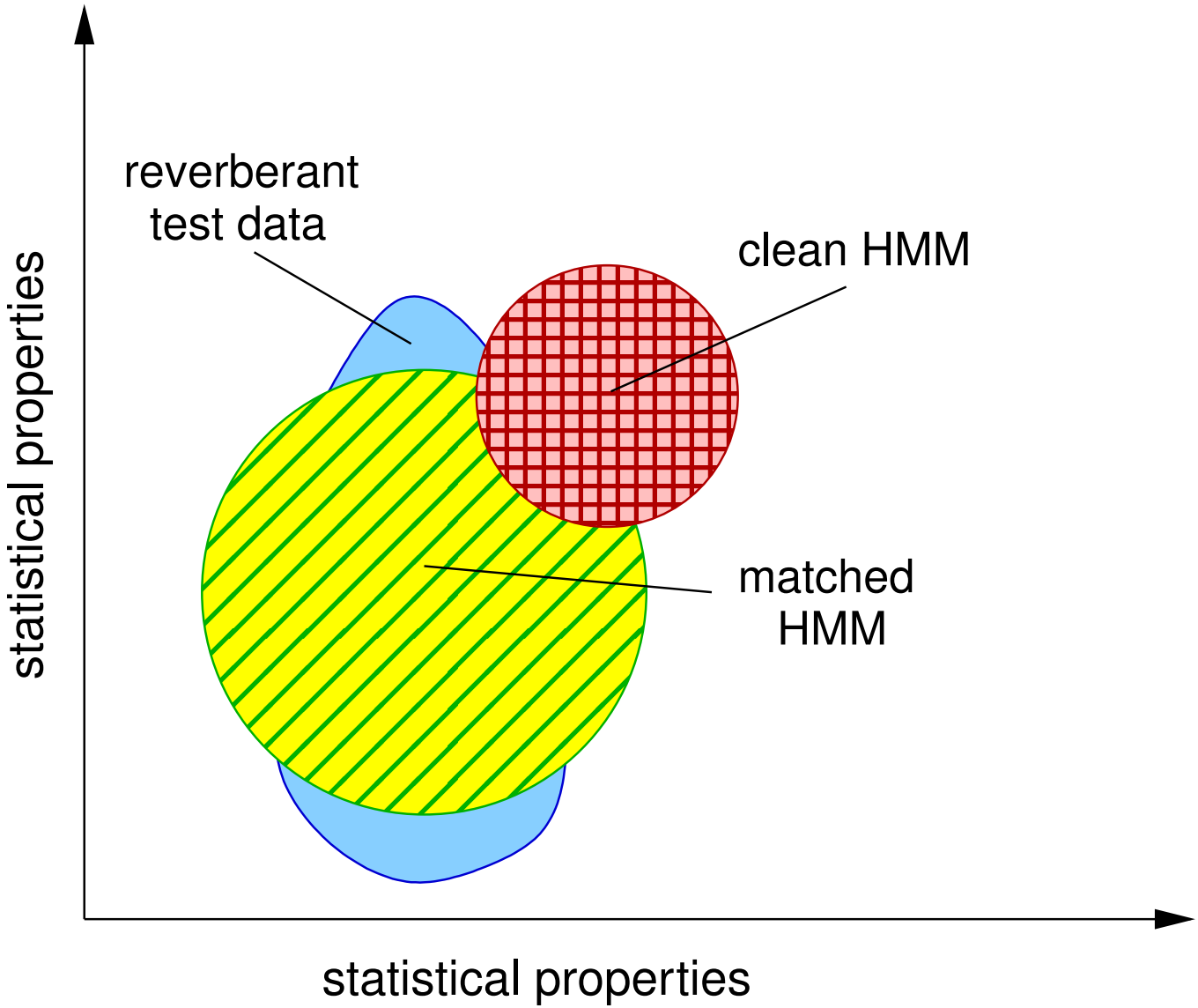
Matched Training

- Record training data in target environment
  - \+ Training data perfectly capture statistical properties
  - – Extremely high effort
- Generate training data by convolution with RIR [Giuliani 1999, Stahl 2001, Matassoni 2002]
  - \+ Significantly reduced effort
  - \+ Only slight degradation in recognition performance [Stahl 2001]

Multi-Style Training

- Use training data from many different rooms
  - \+ Robust HMMs
  - \+ Very flexible
  - – Discrimination capability reduced compared to matched training

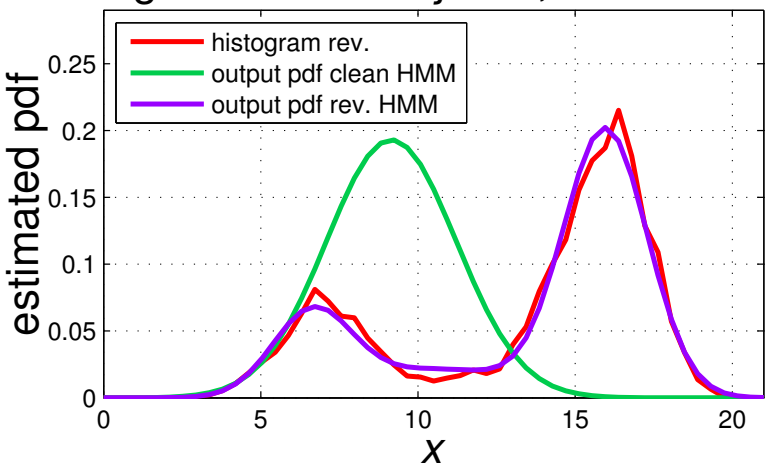
Matched Training



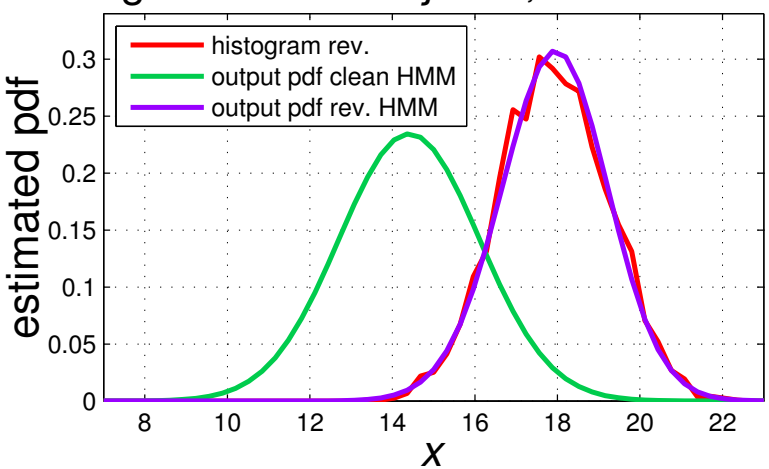
Matched Training: Modeling Accuracy

Histograms

histograms for state $j = 1$, channel $l = 3$



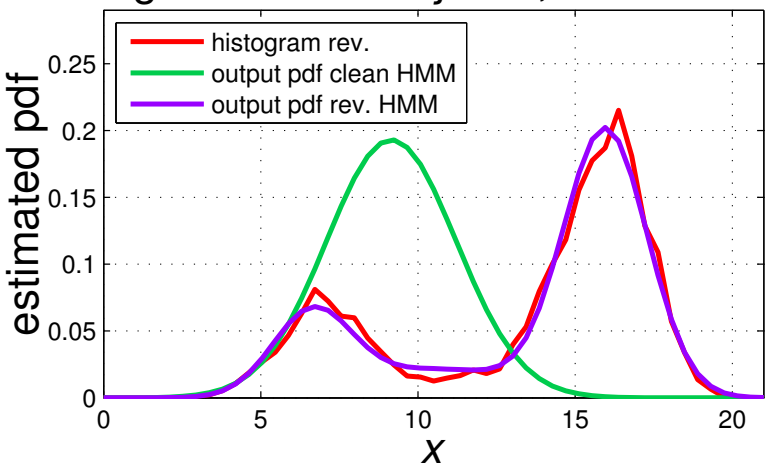
histograms for state $j = 5$, channel $l = 21$



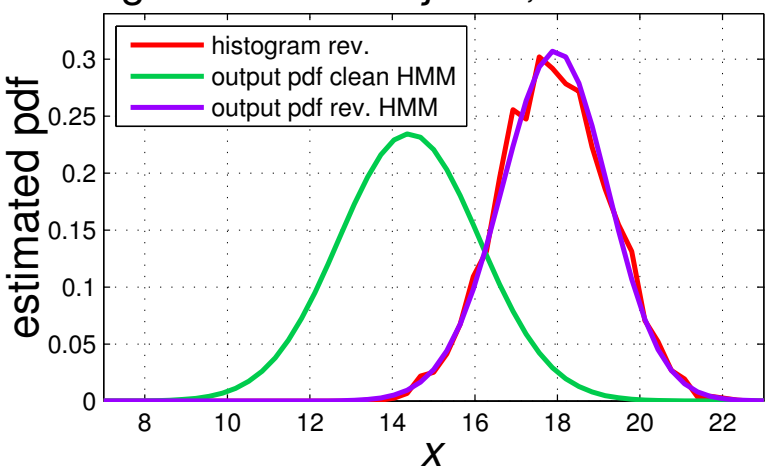
Matched Training: Modeling Accuracy

Histograms

histograms for state $j = 1$, channel $l = 3$

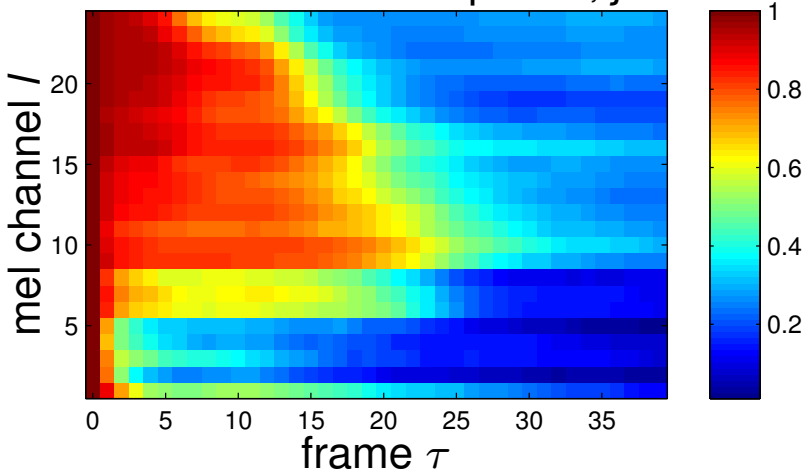


histograms for state $j = 5$, channel $l = 21$

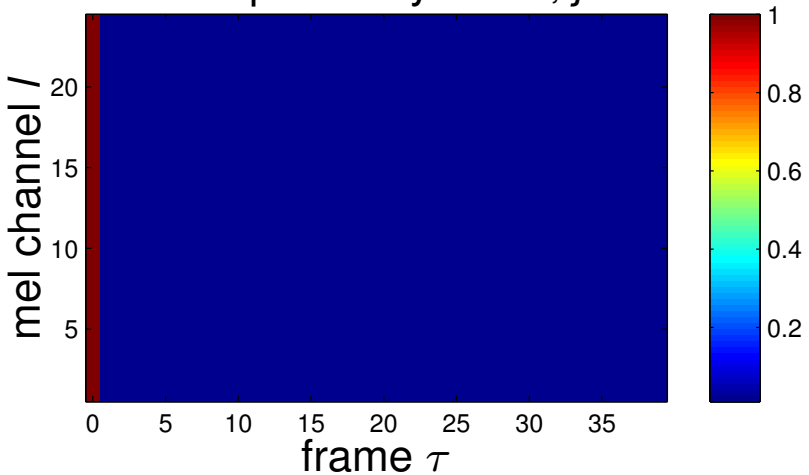


Auto-CoVariances (ACVs)

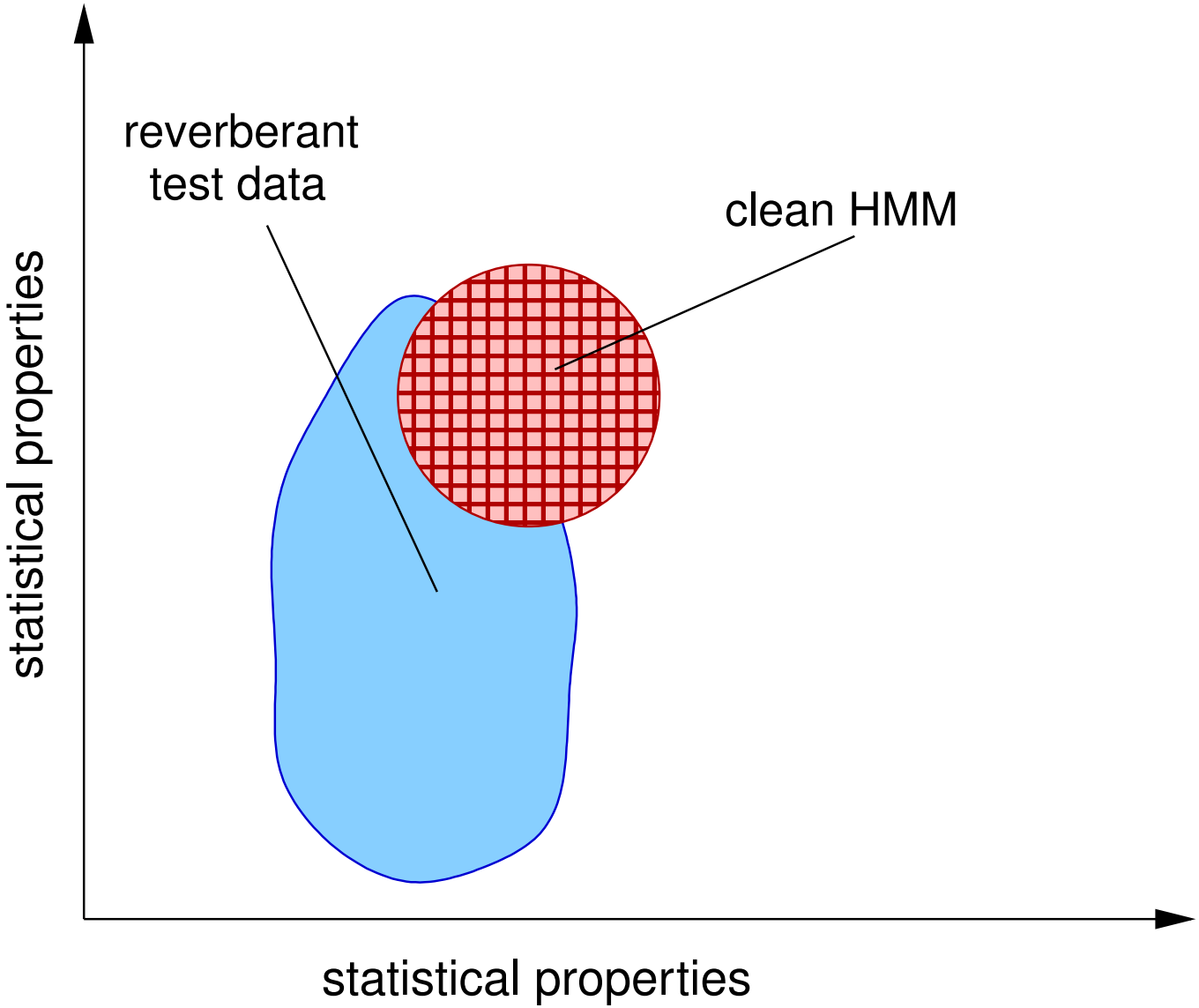
ACVs of reverberant speech, $j = 9$



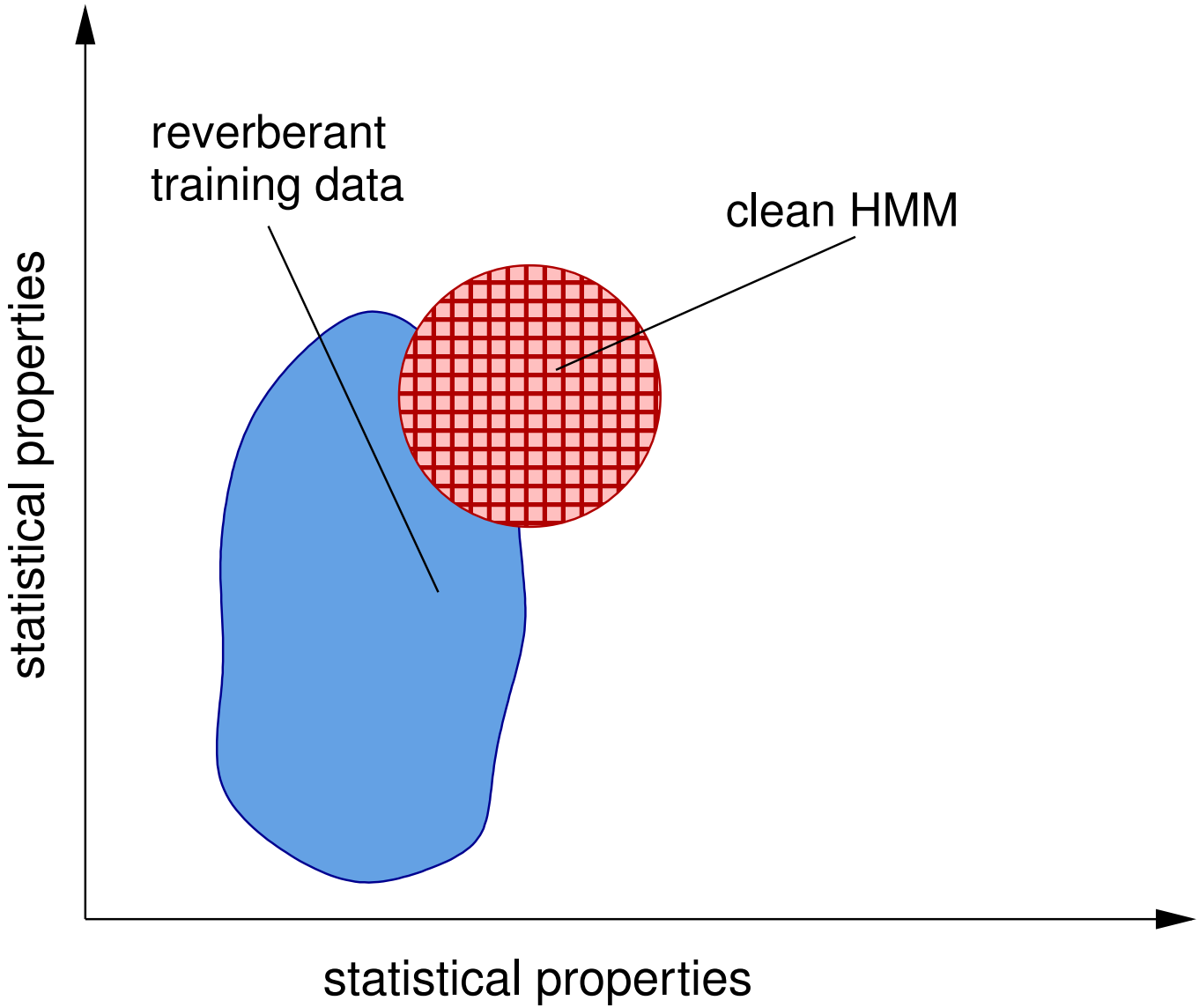
ACVs captured by HMM, $j = 9$



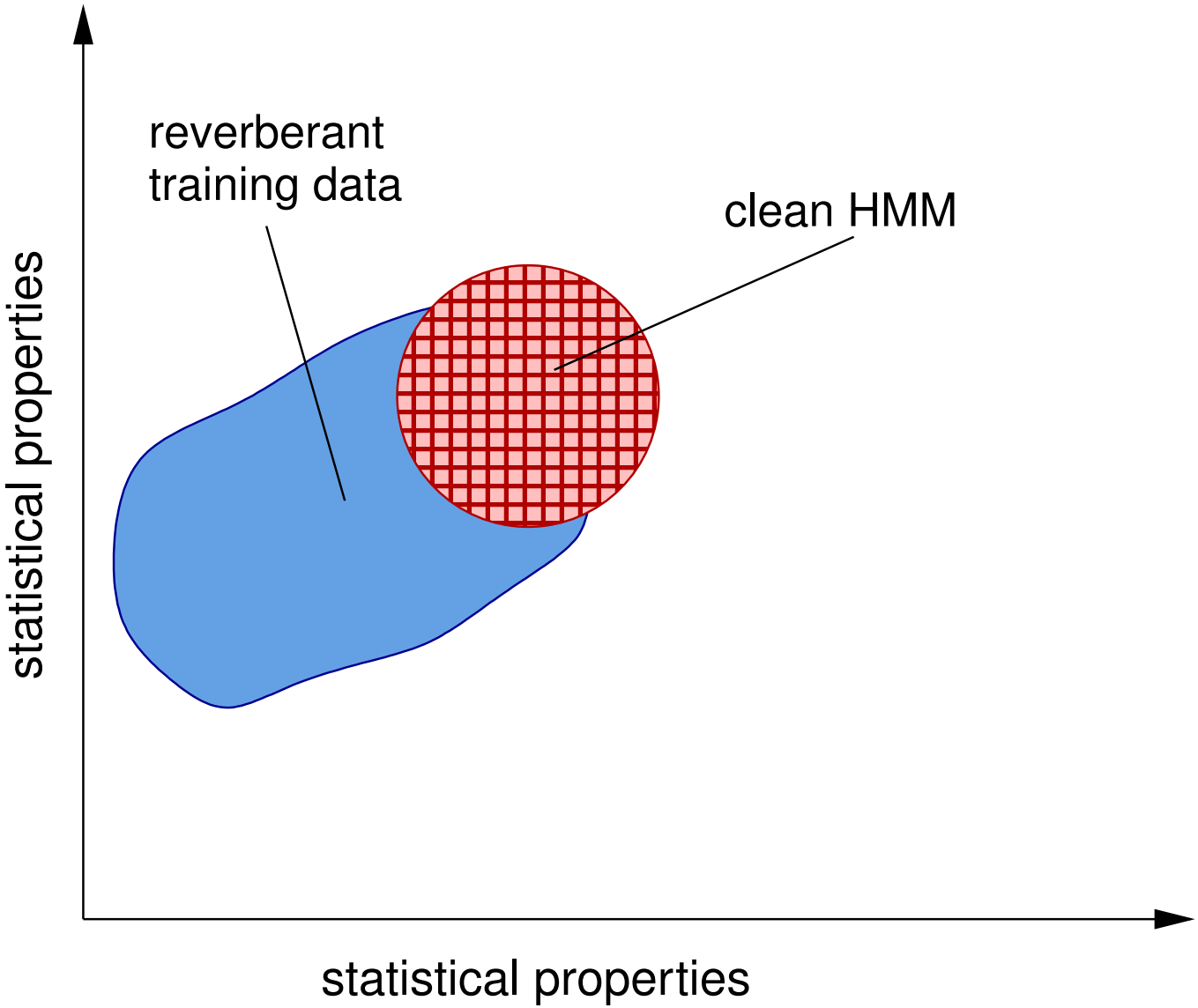
Multi-Style Training



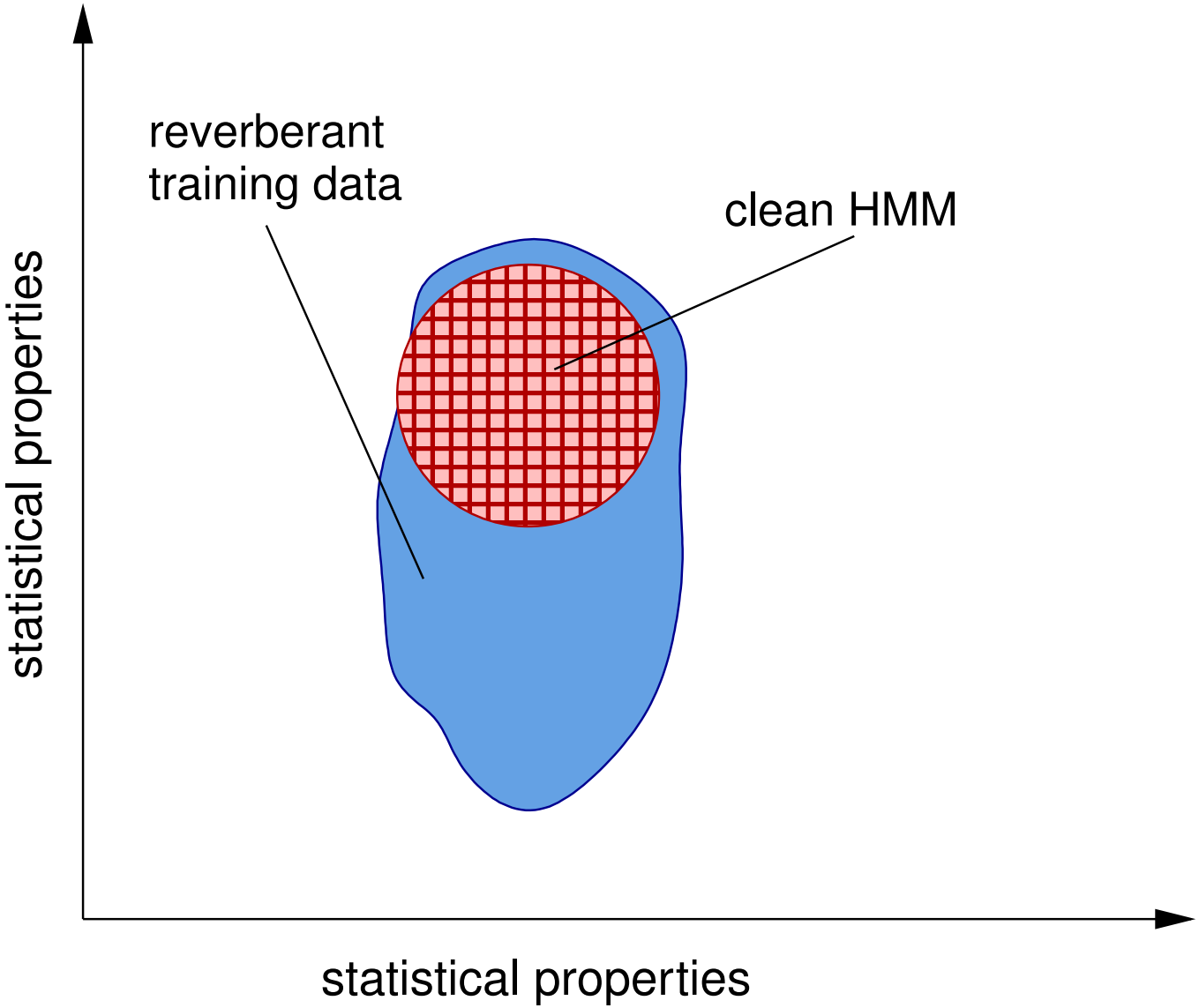
Multi-Style Training



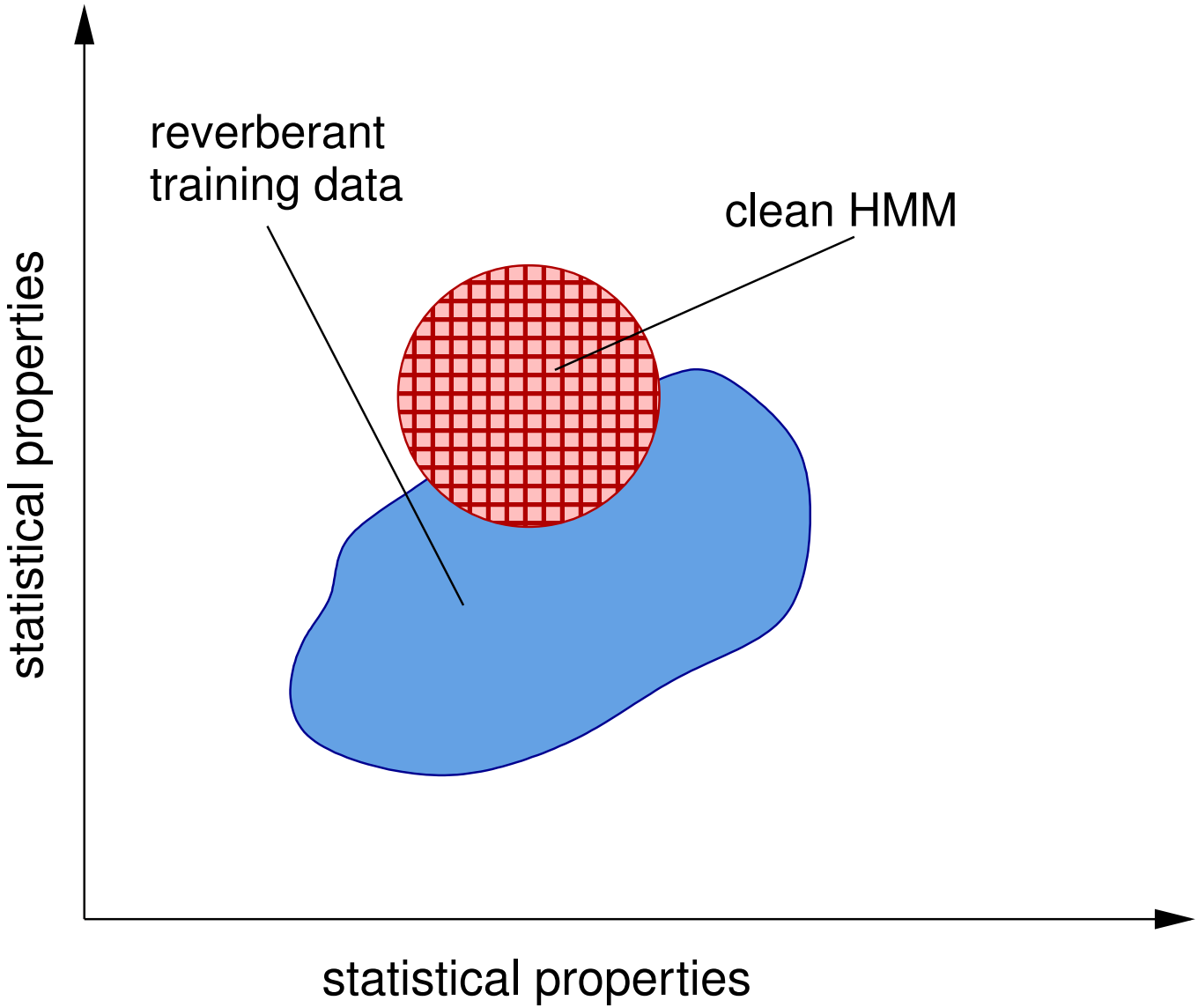
Multi-Style Training



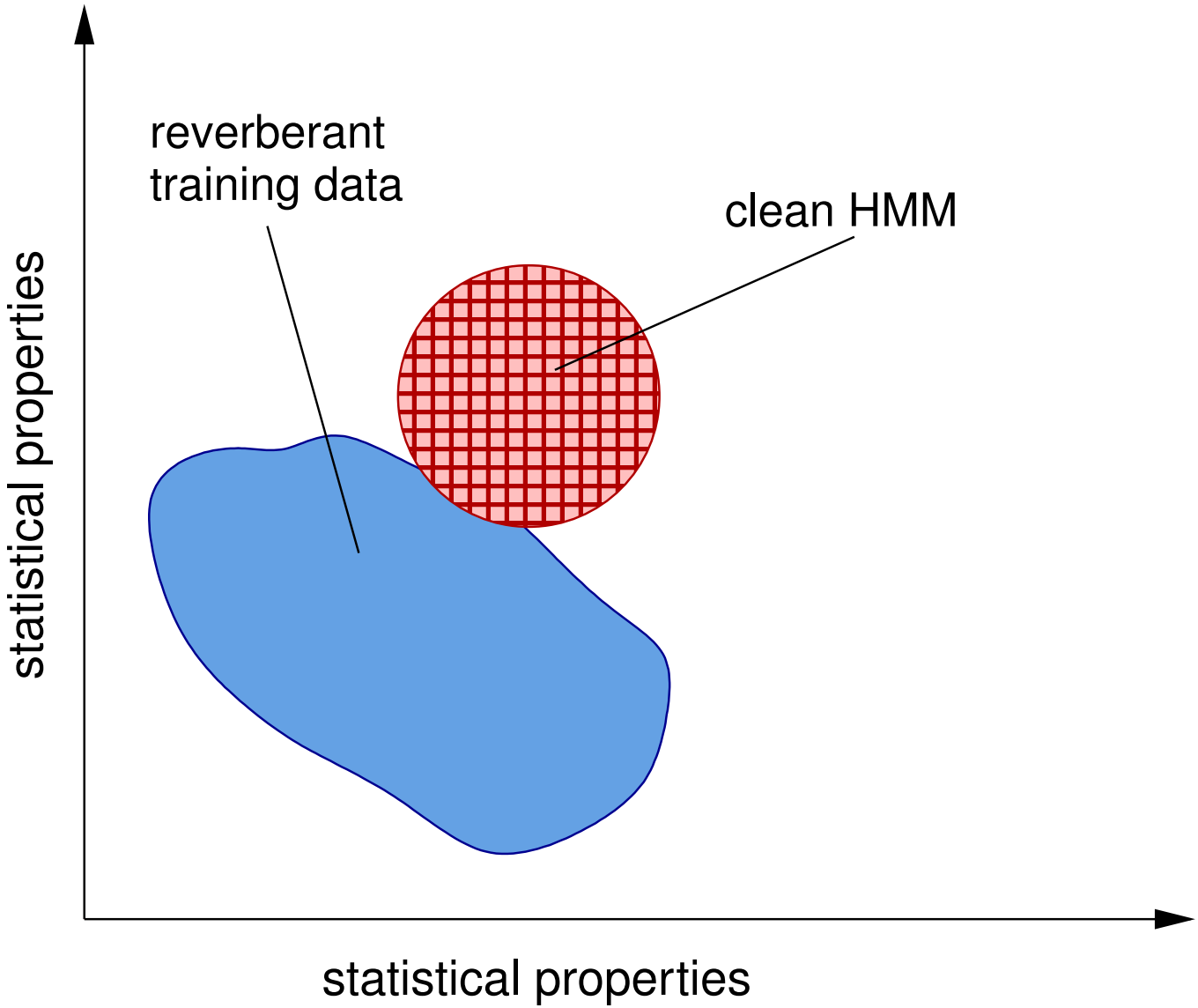
Multi-Style Training



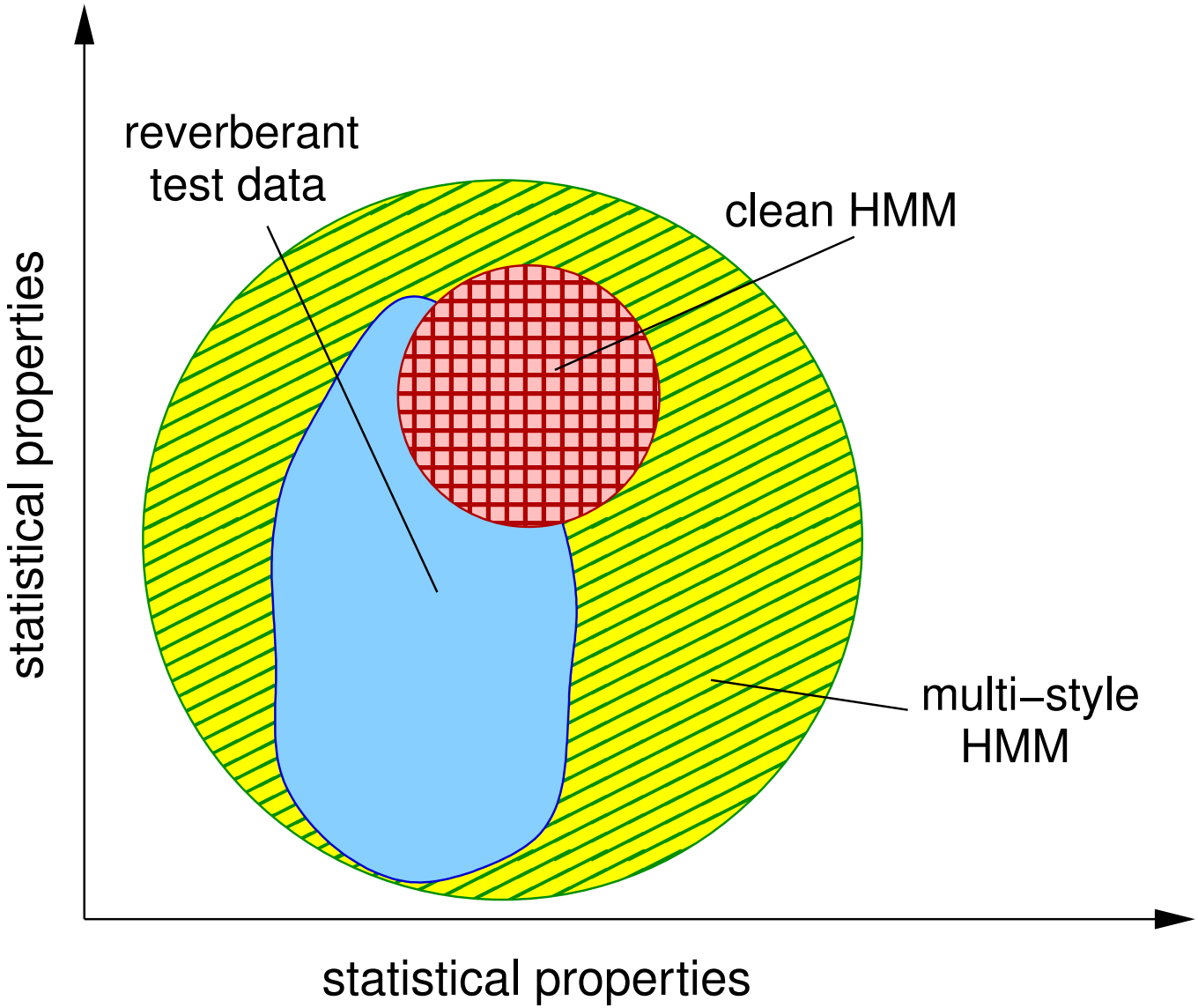
Multi-Style Training



Multi-Style Training



Multi-Style Training



Reverberation-Adaptive Training

Idea

- Capture only linguistic variabilities by acoustic model
- Remove acoustic variabilities by appropriate transforms

Reverberation-Adaptive Training

Idea

- Capture only linguistic variabilities by acoustic model
- Remove acoustic variabilities by appropriate transforms

Approach

- Multi-style training with dereverberated data

Reverberation-Adaptive Training

Idea

- Capture only linguistic variabilities by acoustic model
- Remove acoustic variabilities by appropriate transforms

Approach

- Multi-style training with dereverberated data
- Similar to noise-adaptive training [Deng 2000] or model-independent adaptive training [Gales 2001]

Reverberation-Adaptive Training

Idea

- Capture only linguistic variabilities by acoustic model
- Remove acoustic variabilities by appropriate transforms

Approach

- Multi-style training with dereverberated data
- Similar to noise-adaptive training [Deng 2000] or model-independent adaptive training [Gales 2001]
  - $+$ long-term relations partly removed by dereverberation
  - $+$ room dependency reduced
  - $\Rightarrow$ discrimination capability increased compared to multi-style training

Reverberation-Adaptive Training

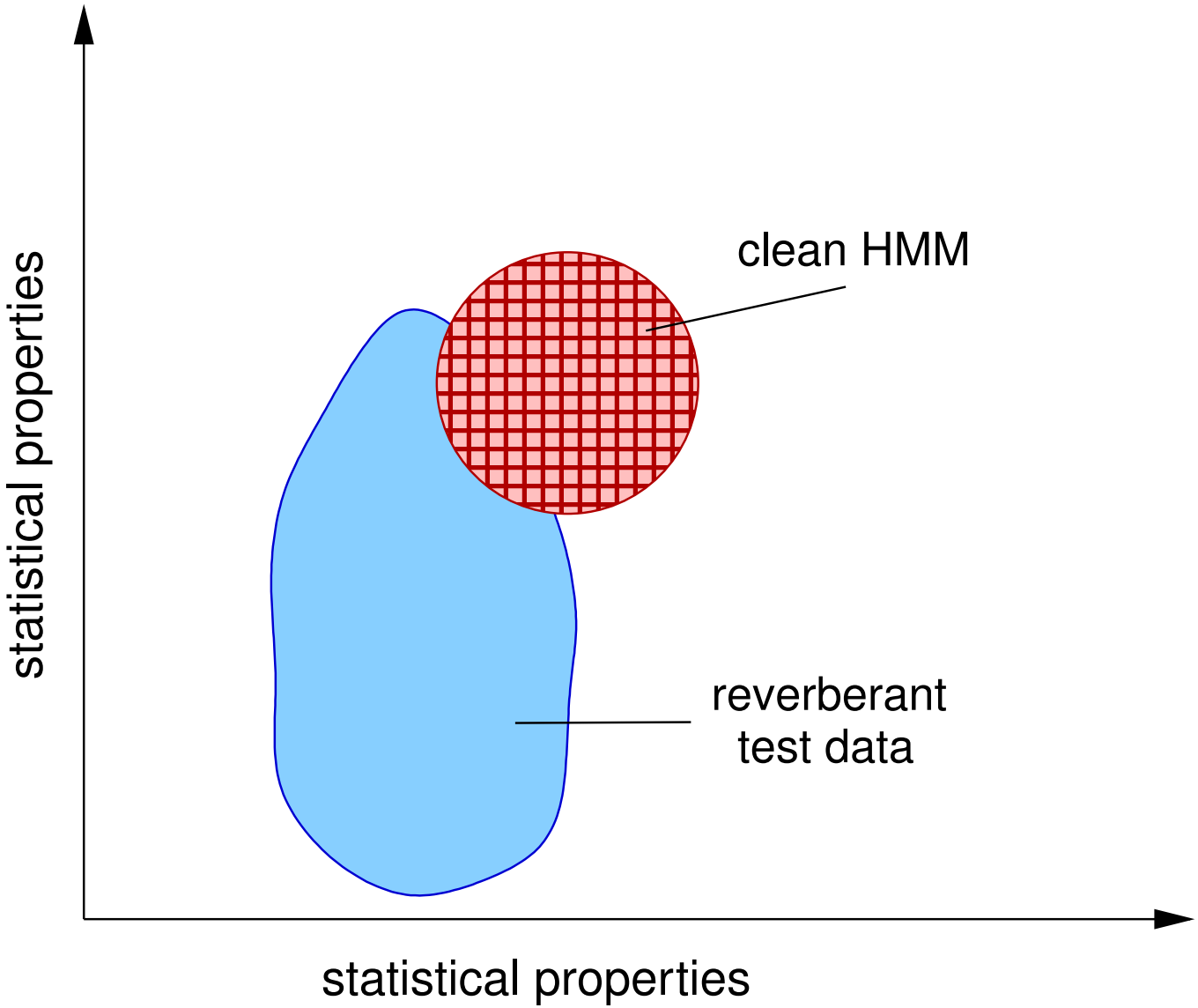
Idea

- Capture only linguistic variabilities by acoustic model
- Remove acoustic variabilities by appropriate transforms

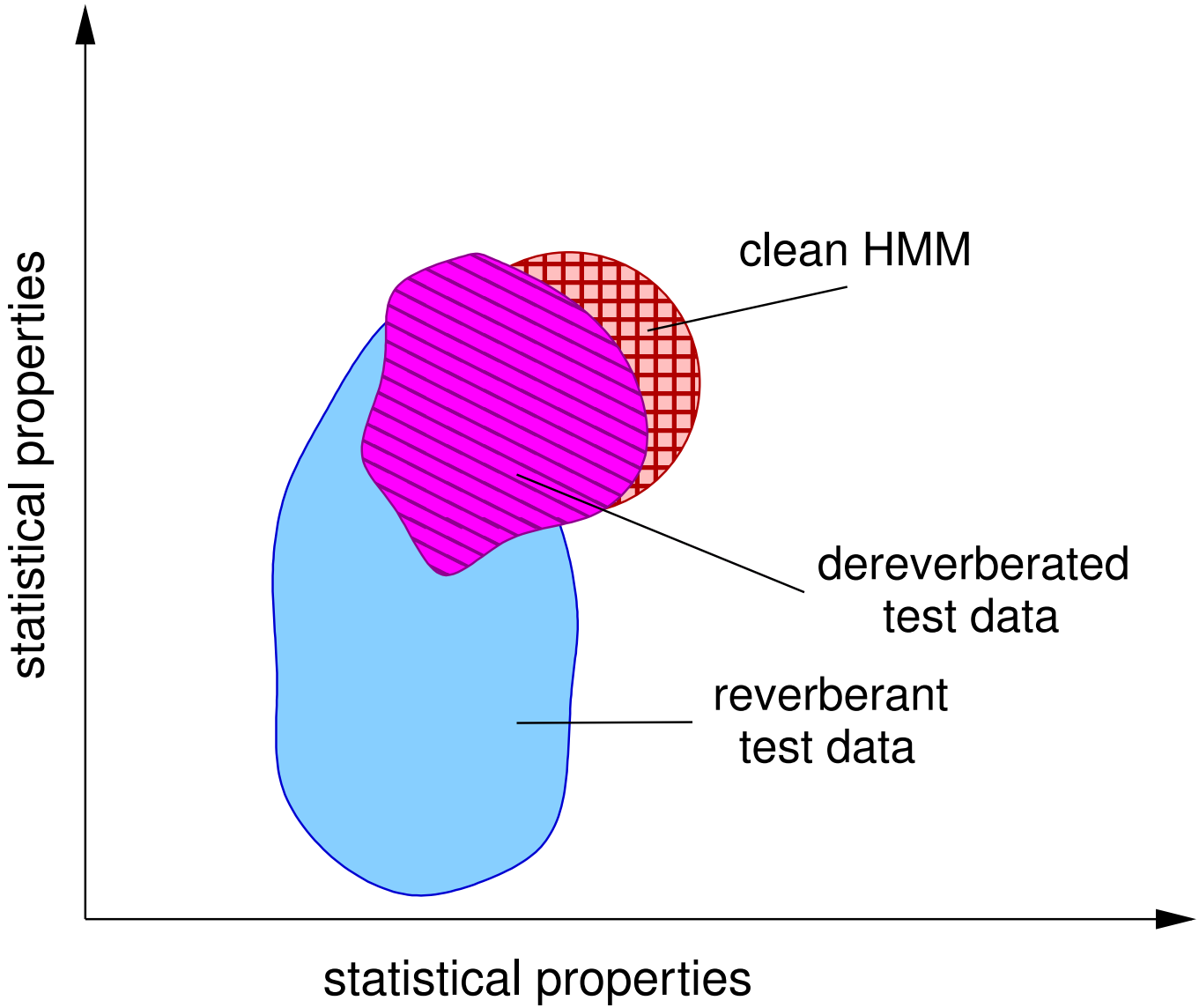
Approach

- Multi-style training with dereverberated data
- Similar to noise-adaptive training [Deng 2000] or model-independent adaptive training [Gales 2001]
  - $+$ long-term relations partly removed by dereverberation
  - $+$ room dependency reduced
  - $\Rightarrow$ discrimination capability increased compared to multi-style training
- Successfully applied, e.g., in [Kinoshita 2006]

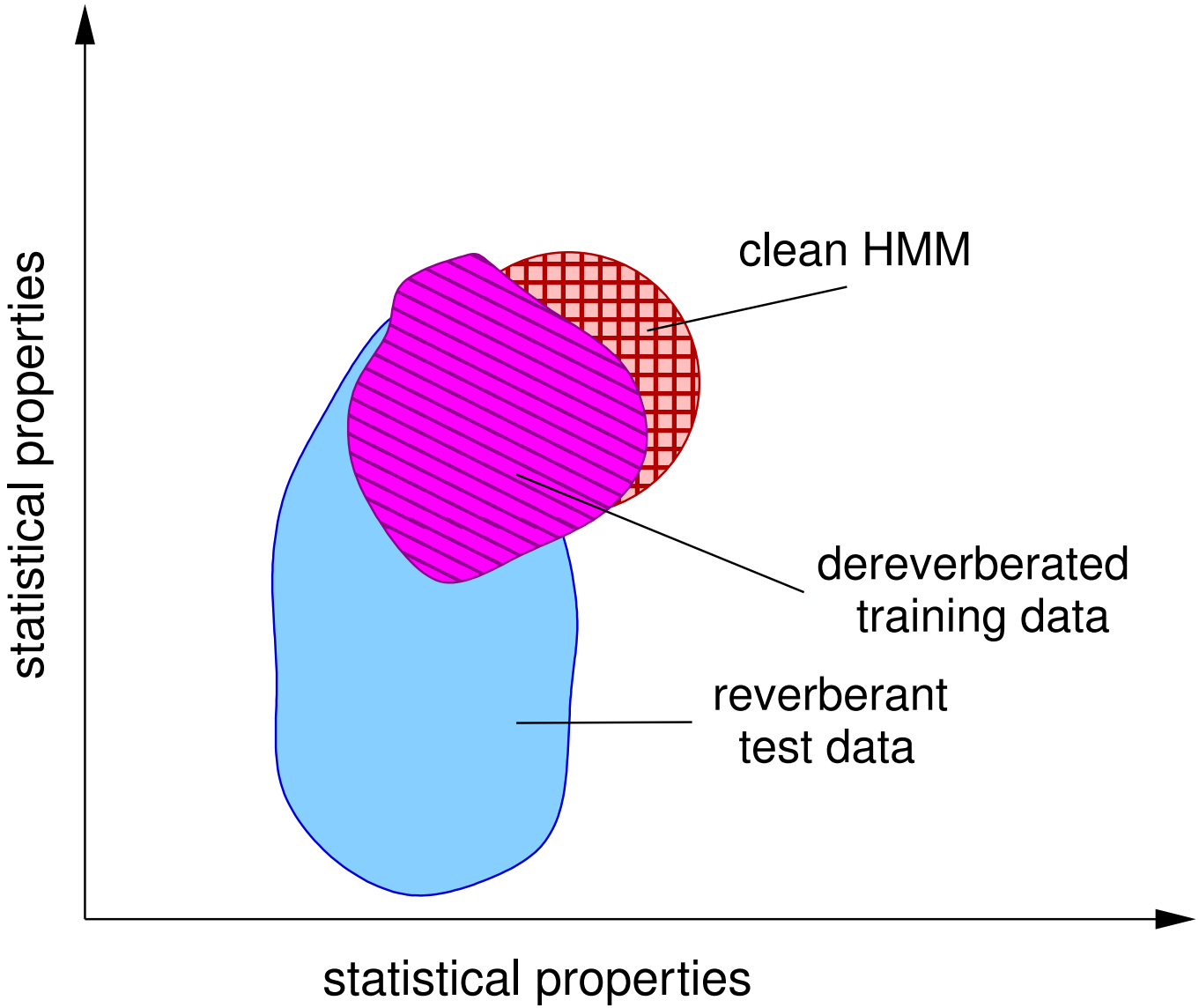
Reverberation-Adaptive Training



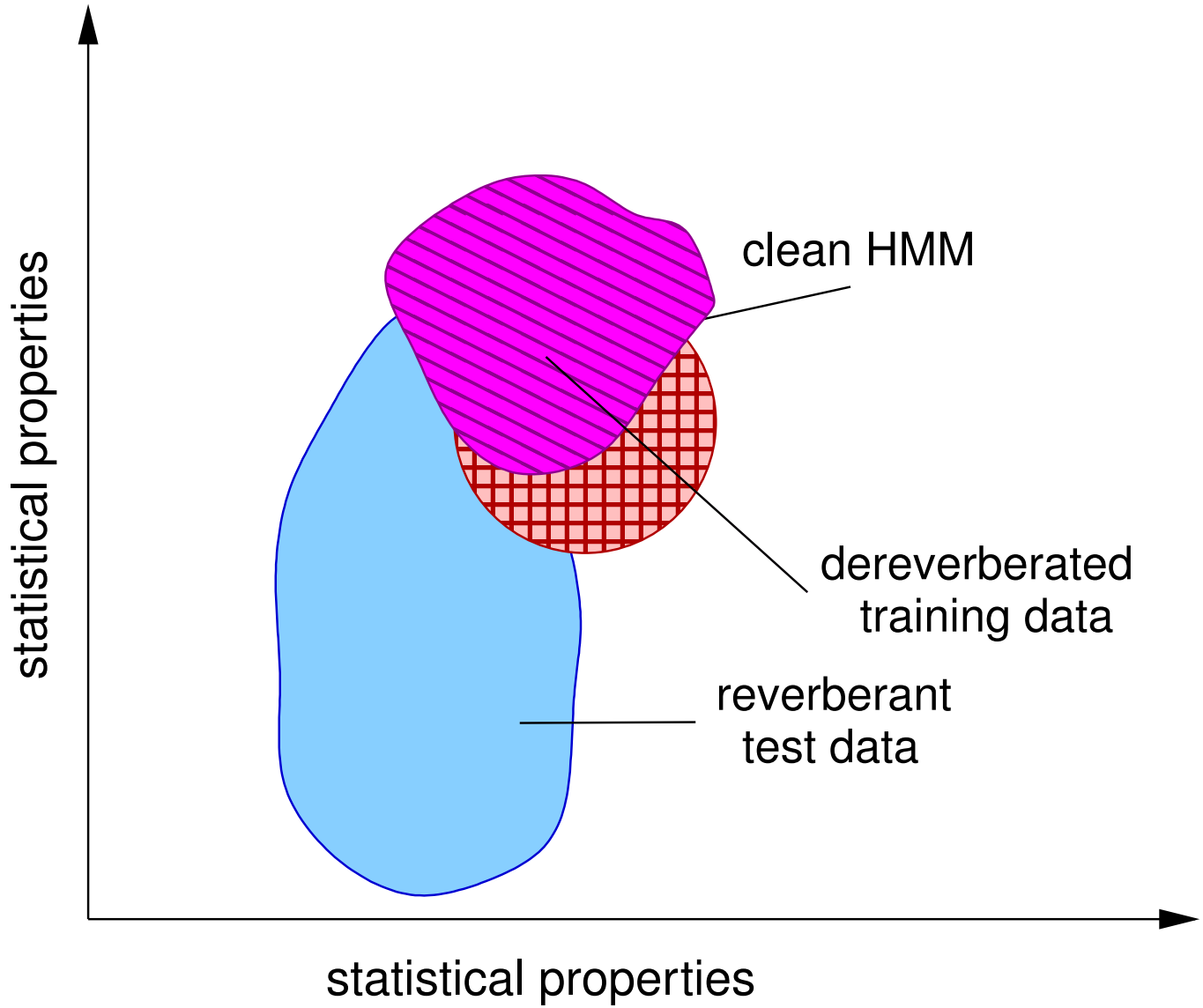
Reverberation-Adaptive Training



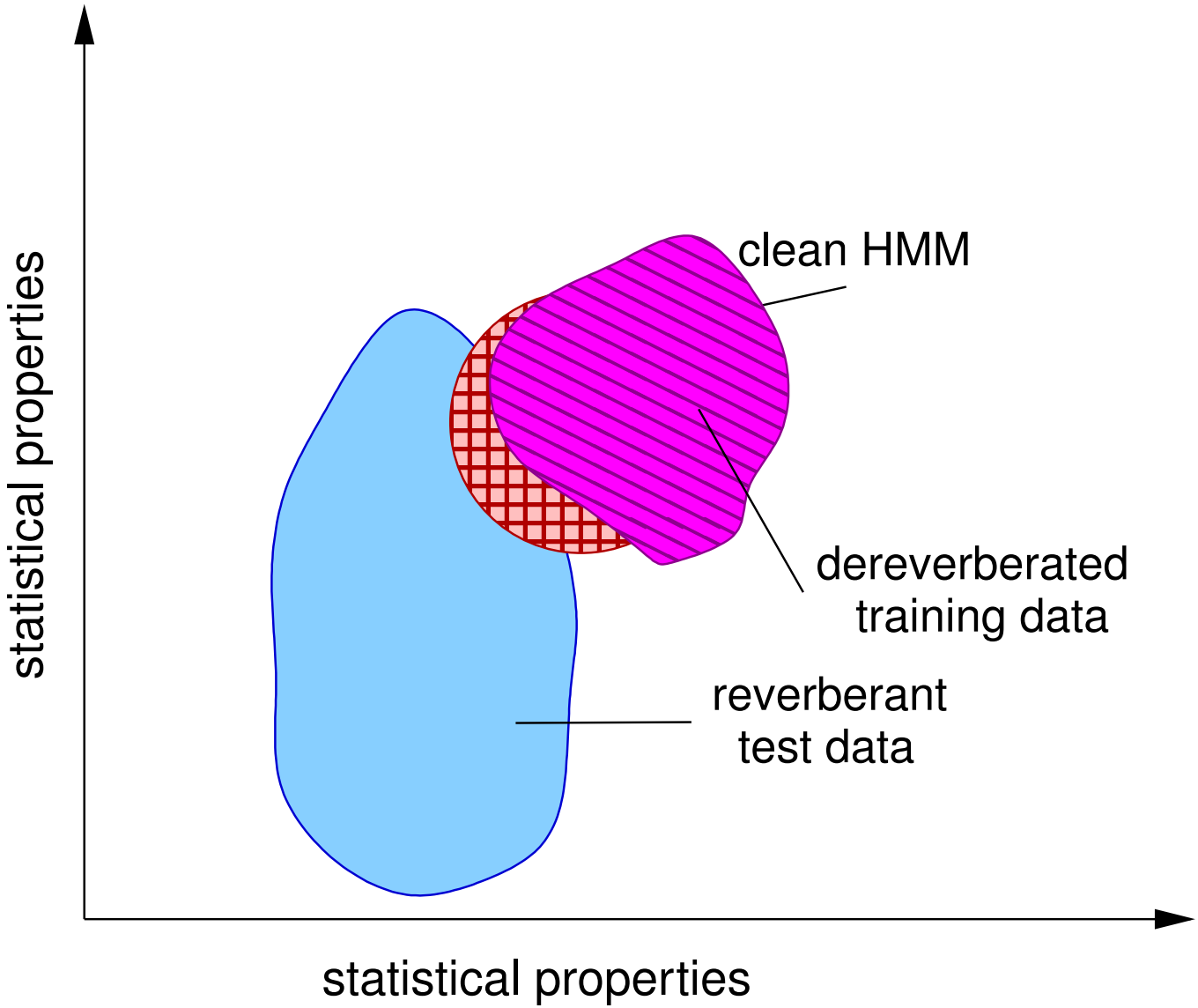
Reverberation-Adaptive Training



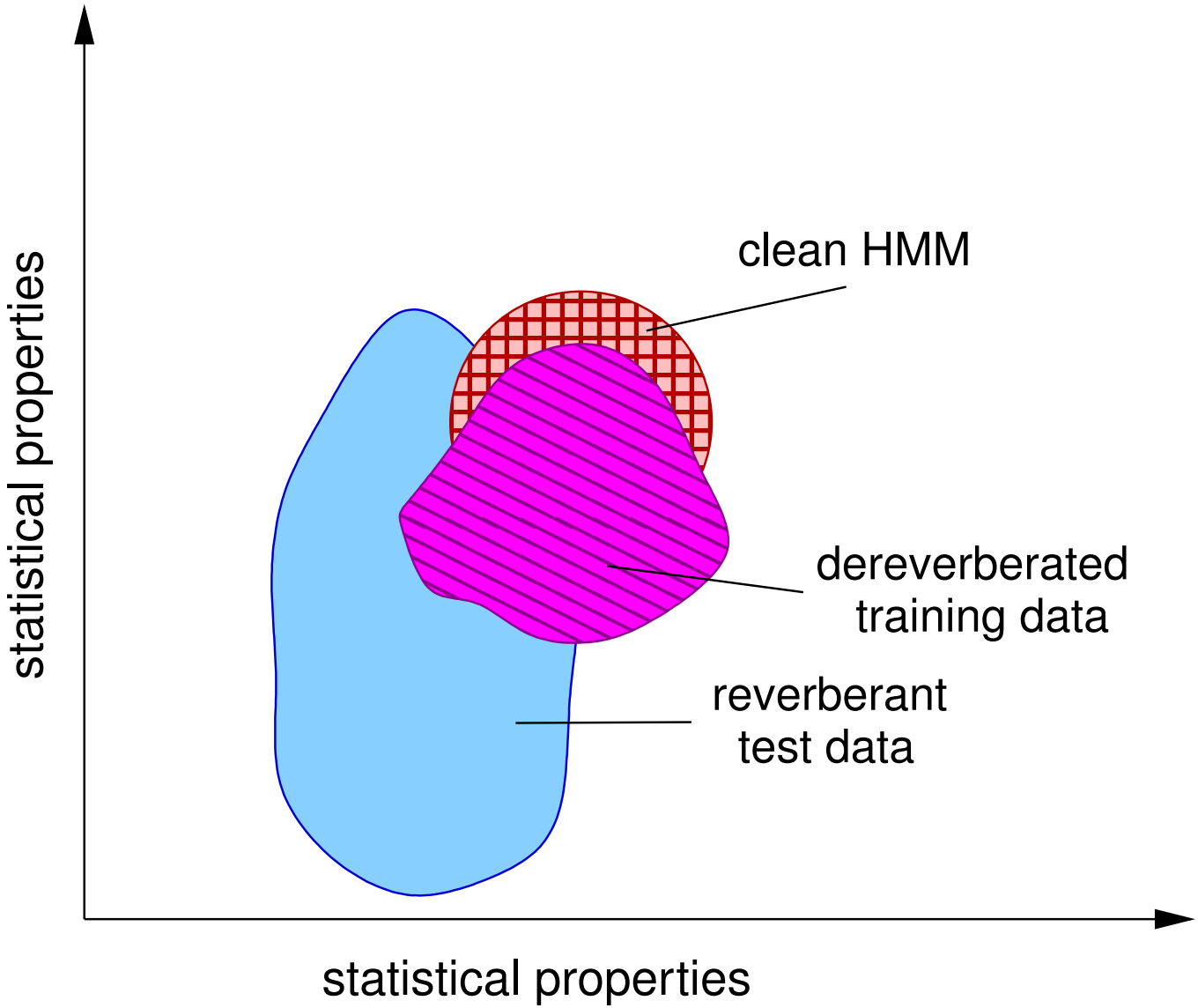
Reverberation-Adaptive Training



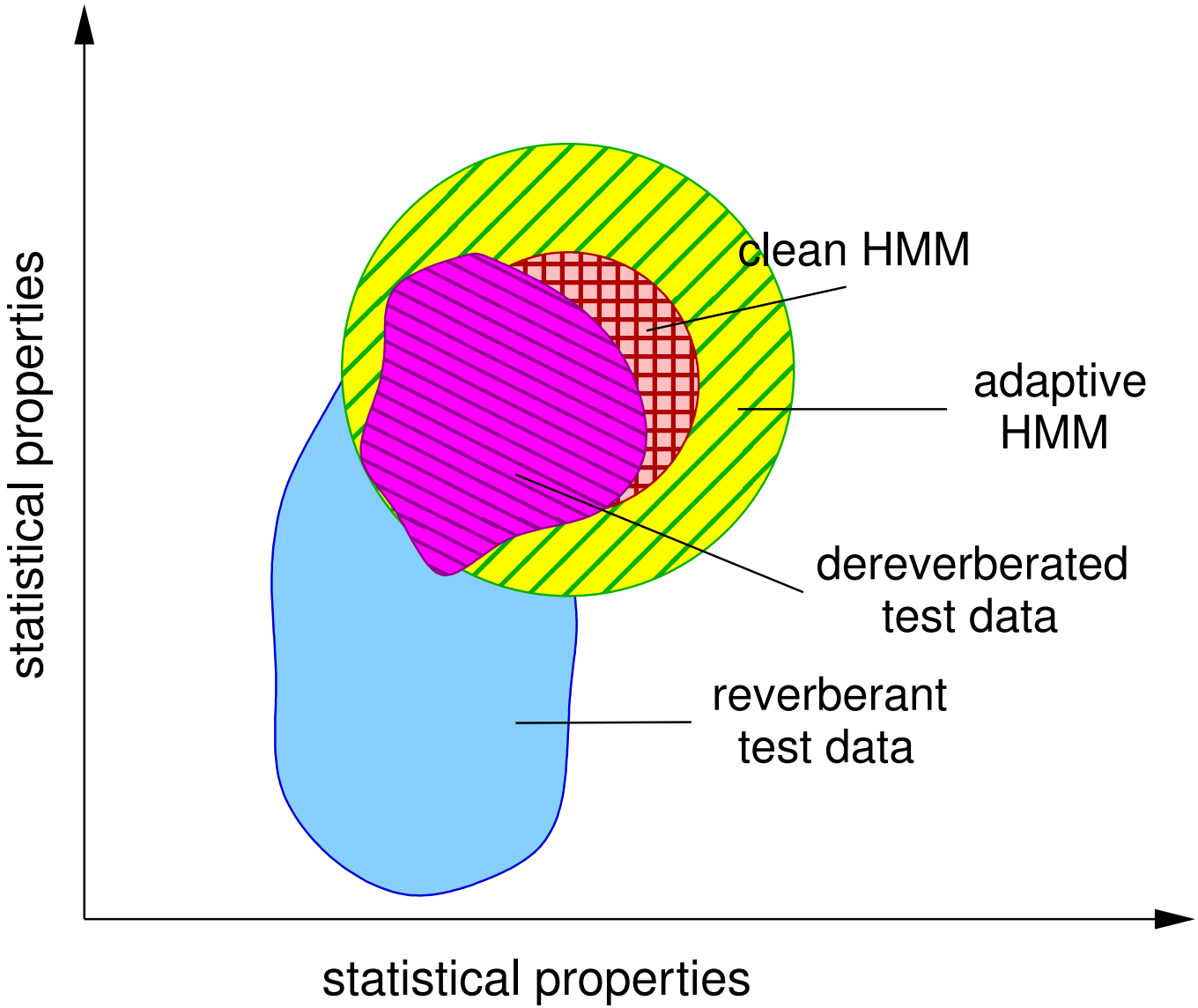
Reverberation-Adaptive Training



Reverberation-Adaptive Training



Reverberation-Adaptive Training



Data-driven Adaptation

Approaches

- Maximum A Posteriori adaptation (MAP) [Gauvain 1994]
- Maximum Likelihood Linear Regression (MLLR) [Legetter 1995, Gales 1998]

Data-driven Adaptation

Approaches

- Maximum A Posteriori adaptation (MAP) [Gauvain 1994]
- Maximum Likelihood Linear Regression (MLLR) [Legetter 1995, Gales 1998]
- Successfully used for speaker and noise adaptation
- Can also be used for reducing mismatch due to reverberation

MLLR

Adaptation of the HMM mean vectors

$$\mu_x = D\mu_s + d$$

MLLR

Adaptation of the HMM mean vectors and covariance matrices

$$\begin{aligned} \mu_{\boldsymbol{x}} &= \boldsymbol{D}\mu_{\boldsymbol{s}} + \boldsymbol{d} \\ \Sigma_{\boldsymbol{xx}} &= \boldsymbol{E}\,\Sigma_{\boldsymbol{ss}}\,\boldsymbol{E}^{T} \end{aligned}$$

MLLR

Adaptation of the HMM mean vectors and covariance matrices

$$\begin{aligned} \boldsymbol{\mu_x} &= \boldsymbol{D}\boldsymbol{\mu_s} + \boldsymbol{d} \\ \boldsymbol{\Sigma_{xx}} &= \boldsymbol{E}\,\boldsymbol{\Sigma_{ss}}\,\boldsymbol{E}^T \end{aligned}$$

Transformation parameters $\boldsymbol{D}$, $\boldsymbol{d}$, $\boldsymbol{E}$ estimated by EM algorithm

MLLR

Adaptation of the HMM mean vectors and covariance matrices

$$\begin{aligned} \boldsymbol{\mu_x} &= \boldsymbol{D}\boldsymbol{\mu_s} + \boldsymbol{d} \\ \boldsymbol{\Sigma_{xx}} &= \boldsymbol{E}\,\boldsymbol{\Sigma_{ss}}\,\boldsymbol{E}^T \end{aligned}$$

Transformation parameters $\boldsymbol{D}$, $\boldsymbol{d}$, $\boldsymbol{E}$ estimated by EM algorithm

- **Supervised MLLR**: known transcription
- **Unsupervised MLLR**: during recognition

MLLR

Adaptation of the HMM mean vectors and covariance matrices

$$\begin{aligned} \mu_{\boldsymbol{x}} &= \boldsymbol{D}\mu_{\boldsymbol{s}} + \boldsymbol{d} \\ \Sigma_{\boldsymbol{xx}} &= \boldsymbol{E}\,\Sigma_{\boldsymbol{ss}}\,\boldsymbol{E}^T \end{aligned}$$

Transformation parameters $\boldsymbol{D}$, $\boldsymbol{d}$, $\boldsymbol{E}$ estimated by EM algorithm

- **Supervised MLLR**: known transcription
- **Unsupervised MLLR**: during recognition

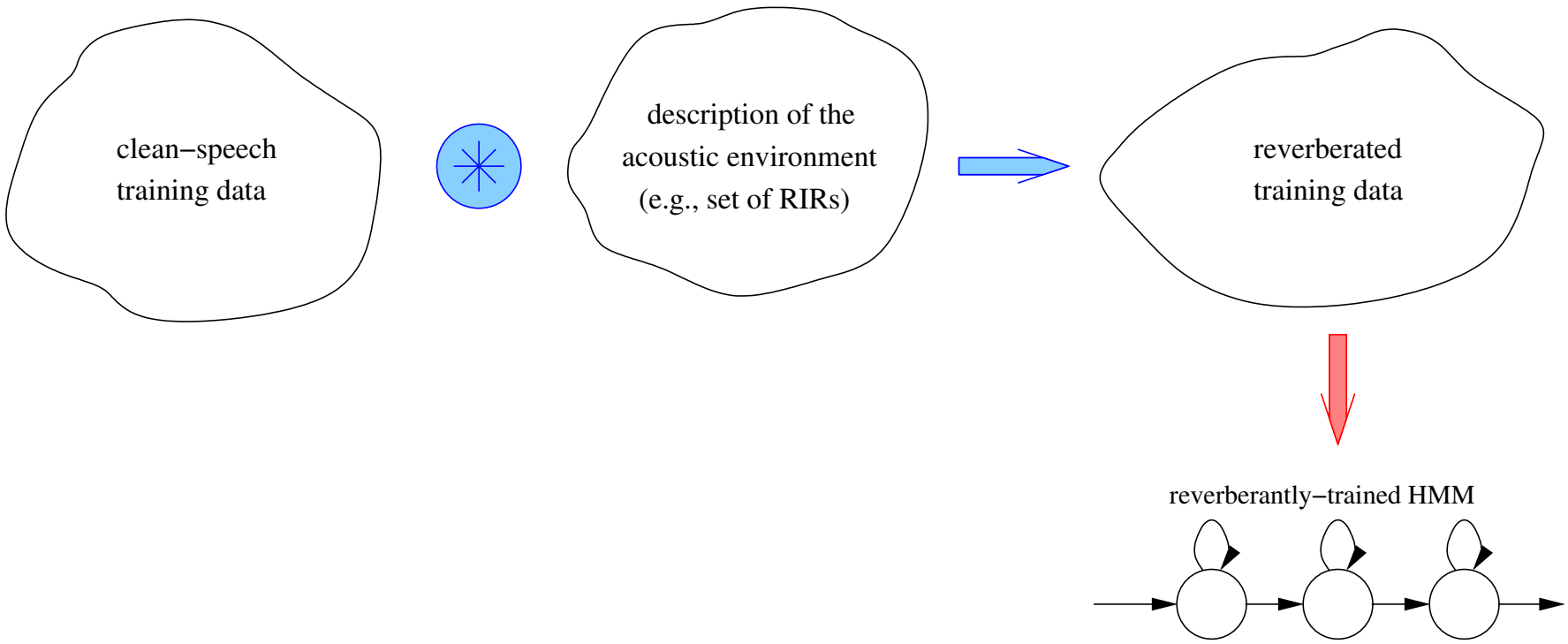
CMLLR (Constrained MLLR)

Same transformation matrix for mean vector and covariance matrix

$$\begin{aligned} \mu_{\boldsymbol{x}} &= \boldsymbol{D}\mu_{\boldsymbol{s}} + \boldsymbol{d} \\ \Sigma_{\boldsymbol{xx}} &= \boldsymbol{D}\,\Sigma_{\boldsymbol{ss}}\,\boldsymbol{D}^T \end{aligned}$$

\+ fewer adaptation parameters

Illustration: Example Matched Training on Reverberated Data



Data-driven Approaches

Discussion

- Very accurate description of statistical properties by reverberant training/adaptation data
- Loss of accuracy: only when turning data into model

Data-driven Approaches

Discussion

- Very accurate description of statistical properties by reverberant training/adaptation data
- Loss of accuracy: only when turning data into model
- Reverberant training: requires large amount of reverberant data

Data-driven Approaches

Discussion

- Very accurate description of statistical properties by reverberant training/adaptation data
- Loss of accuracy: only when turning data into model
- Reverberant training: requires large amount of reverberant data
- Data-driven adaptation: moderate amount of reverberant data (but more than model-based adaptation)

Data-driven Approaches

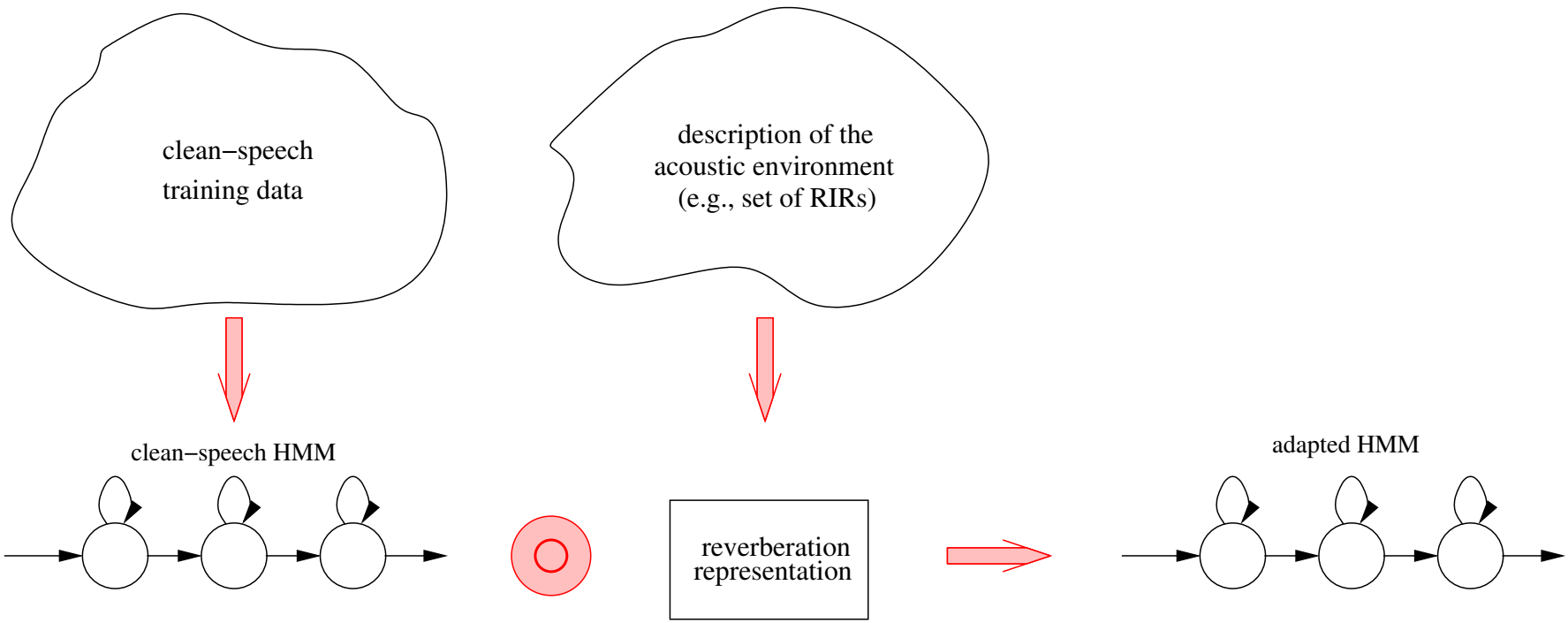
Discussion

- Very accurate description of statistical properties by reverberant training/adaptation data
- Loss of accuracy: only when turning data into model
- Reverberant training: requires large amount of reverberant data
- Data-driven adaptation: moderate amount of reverberant data (but more than model-based adaptation)

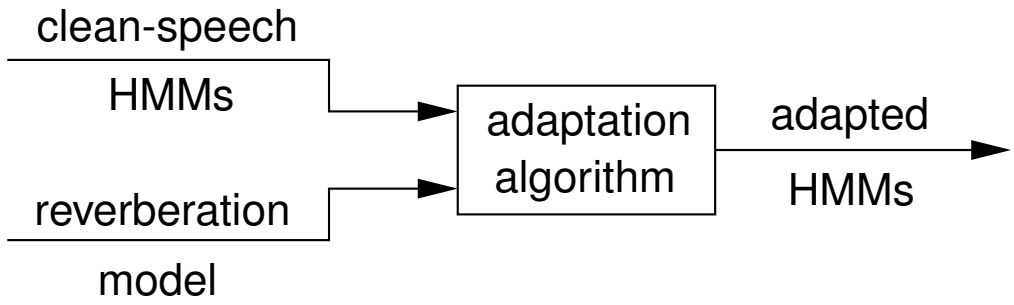
Main Limitation

Conventional HMMs cannot accurately capture long-term relations

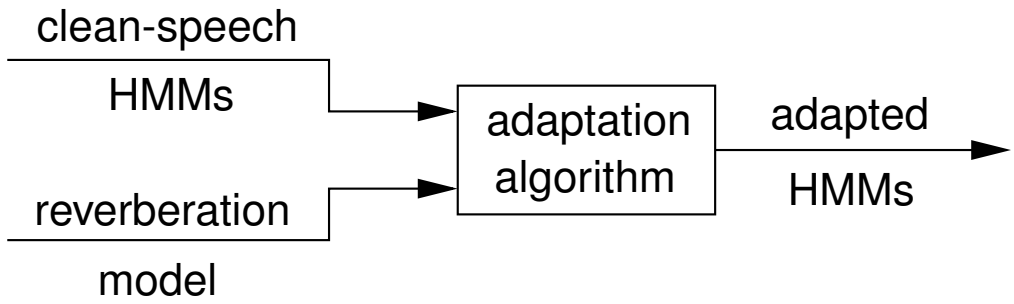
Illustration



Parametric Model-based Adaptation



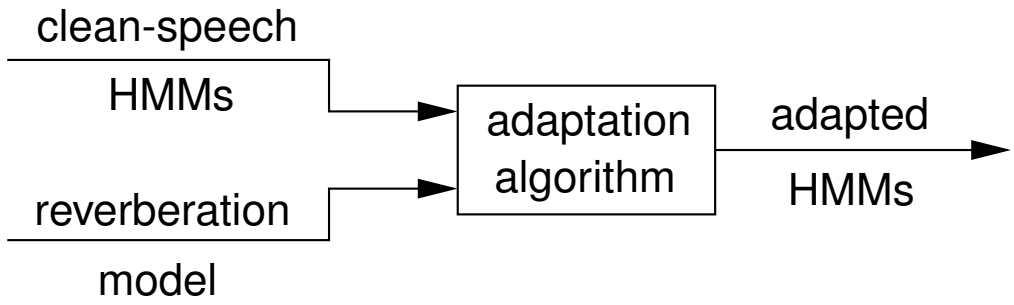
Parametric Model-based Adaptation



Discussion

- proposed in [Raut 2006, Hirsch 2008, Sehr 2009]
- based on melspec convolution

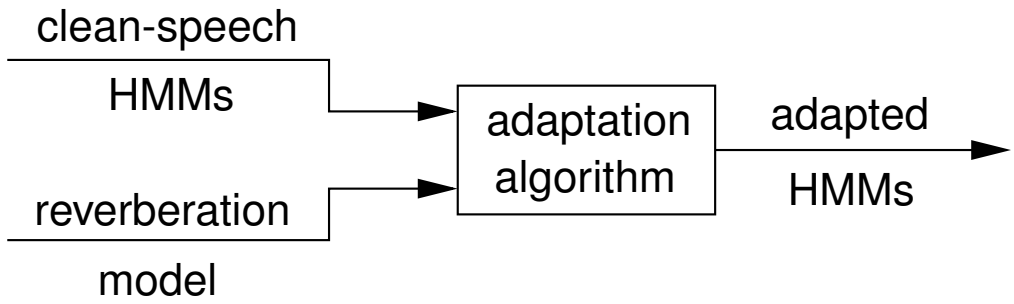
Parametric Model-based Adaptation



Discussion

- proposed in [Raut 2006, Hirsch 2008, Sehr 2009]
- based on melspec convolution
  - \+ long-term relations considered for HMM parameter estimation
  - \+ no adaptation utterances necessary

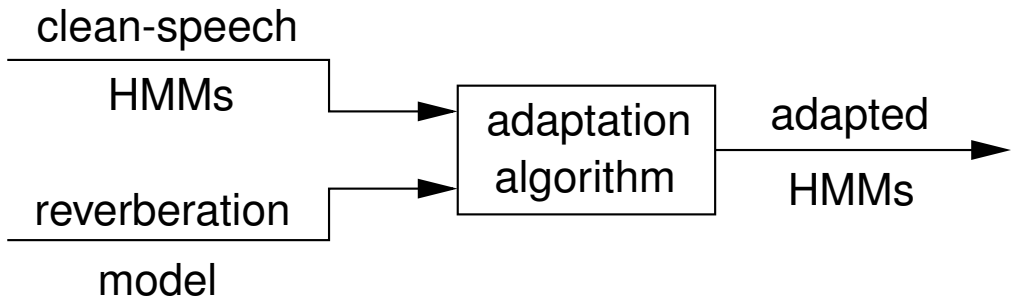
Parametric Model-based Adaptation



Discussion

- proposed in [Raut 2006, Hirsch 2008, Sehr 2009]
- based on melspec convolution
  - \+ long-term relations considered for HMM parameter estimation
  - \+ no adaptation utterances necessary
  - – reduced accuracy due to approximation errors
  - – additional loss of accuracy when mapping combination to HMM

Parametric Model-based Adaptation



Discussion

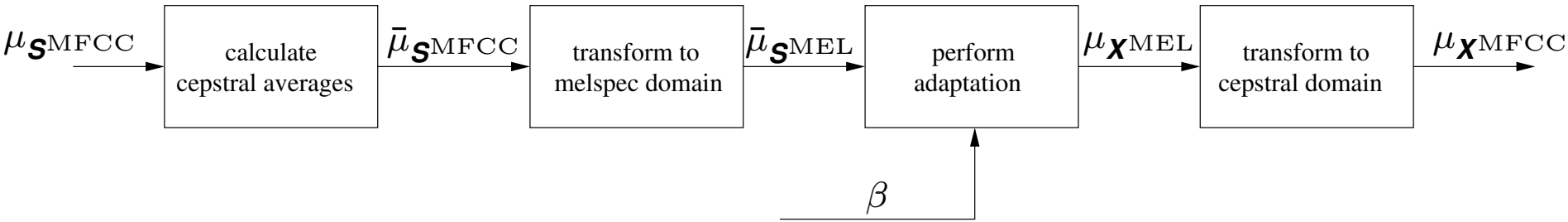
- proposed in [Raut 2006, Hirsch 2008, Sehr 2009]
- based on melspec convolution
  - \+ long-term relations considered for HMM parameter estimation
  - \+ no adaptation utterances necessary
  - – reduced accuracy due to approximation errors
  - – additional loss of accuracy when mapping combination to HMM

Main Limitation

Conventional HMMs cannot accurately capture long-term relations

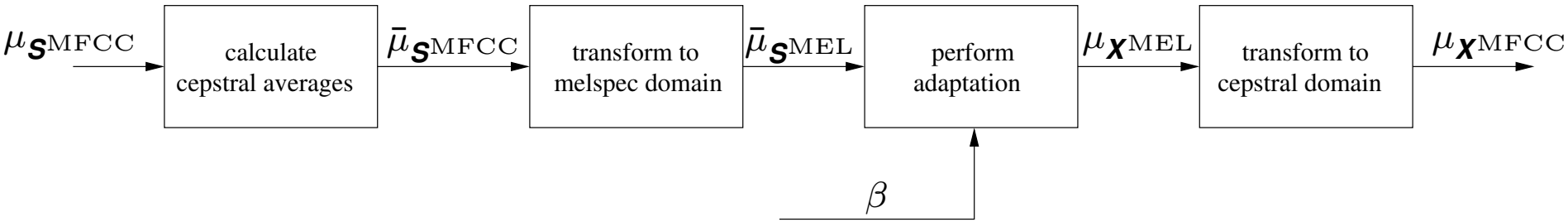
Parametric Model-based Adaptation

Mean Adaptation Approach [Raut 2006, Hirsch 2008, Sehr 2009]



Parametric Model-based Adaptation

Mean Adaptation Approach [Raut 2006, Hirsch 2008, Sehr 2009]

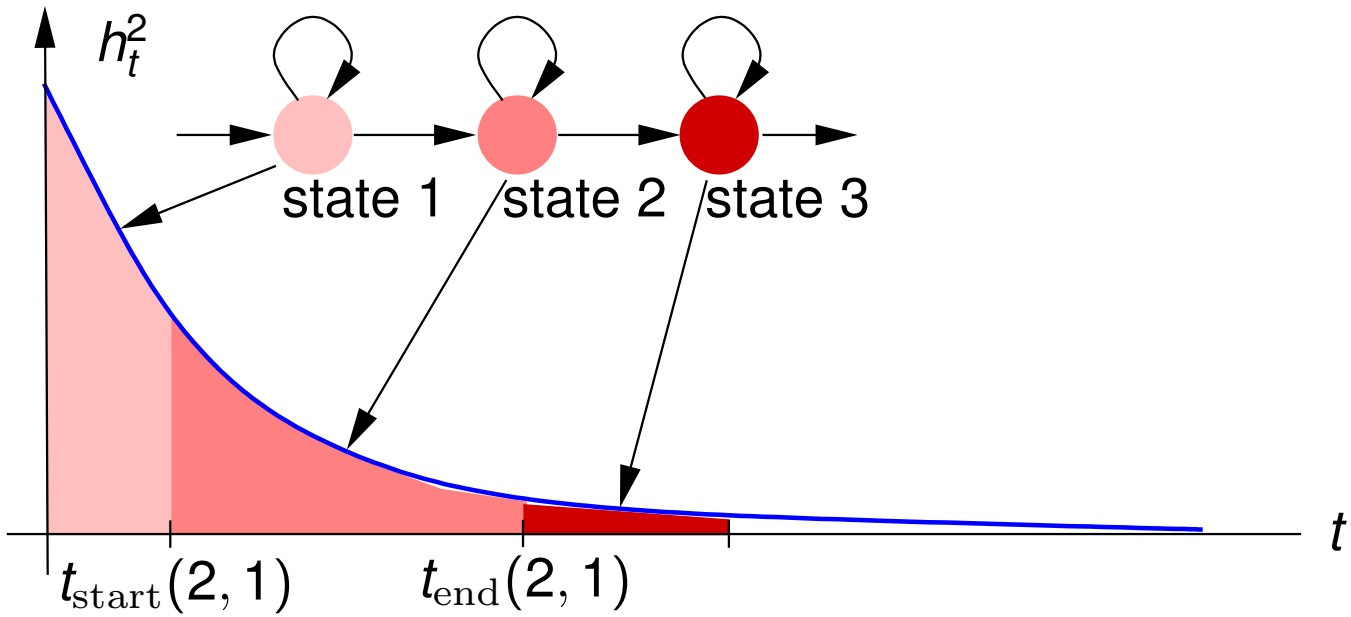


Adaptation Equation

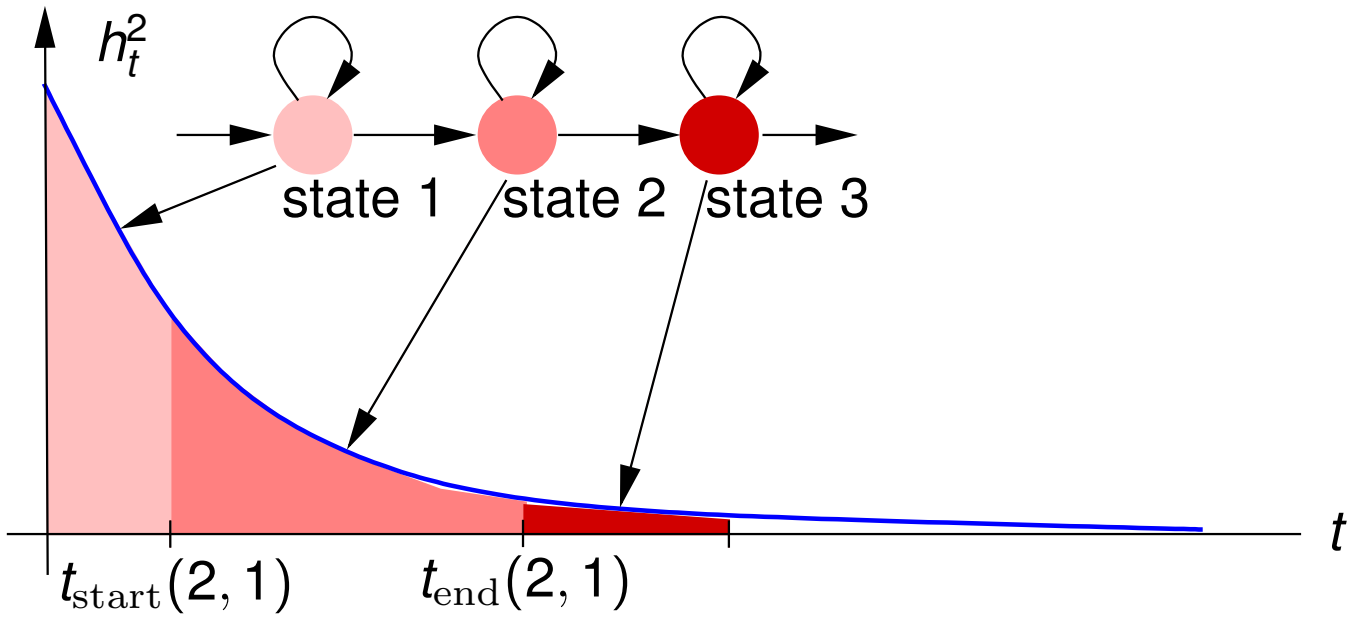
$$\mu_{x^{\mathrm{MEL}}}(l,j) \quad = \quad \sum_{p} \beta(l,j,j-p)\ \mu_{s^{\mathrm{MEL}}}(l,j-p)$$

$\beta(l,j,i)$	state-level reverberation representation: describes energy dispersion from state i to j in channel l
i, j	state indices
l	mel channel index

Estimation of Reverberation Representation [Hirsch 2008]

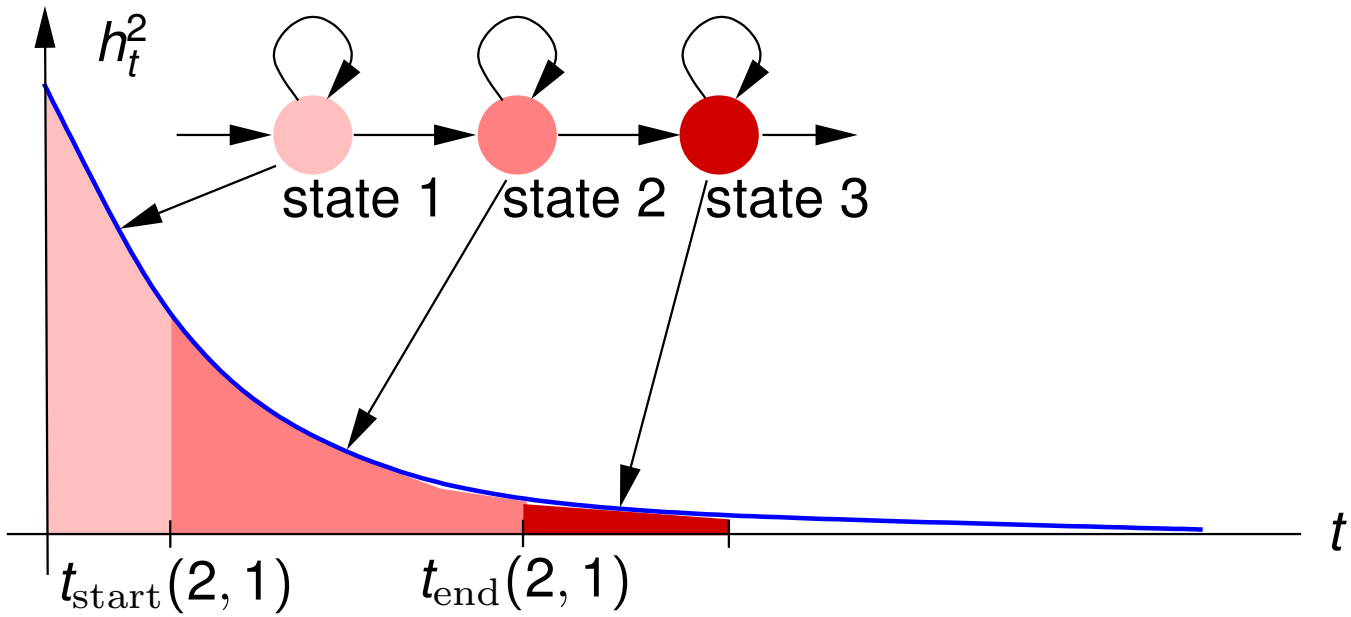


Estimation of Reverberation Representation [Hirsch 2008]



$$h_t^2 = \frac{6\ \log(10)}{T_{60M}} \cdot \exp\left(-\frac{6\ \log(10)}{T_{60M}} \cdot t\right), \quad \text{for} \quad t \geq 0$$

Estimation of Reverberation Representation [Hirsch 2008]



$$h_t^2 = \frac{6\ \log(10)}{T_{60M}} \cdot \exp\left(-\frac{6\ \log(10)}{T_{60M}} \cdot t\right), \quad \text{for} \quad t \geq 0$$

$$\beta(j,i) = \int_{t_{\mathrm{start}}(j,i)}^{t_{\mathrm{end}}(j,i)} h_t^2 \, \mathrm{d}t$$

Limitation of Conventional HMMs

Emission pdf of Conventional HMMs

$$p(\boldsymbol{x}_n|j)$$

$\Rightarrow$ conditional independence assumption

Limitation of Conventional HMMs

Emission pdf of Conventional HMMs

$$p(\boldsymbol{x}_n|j)$$

$\Rightarrow$ conditional independence assumption

Conditional Emission pdf

capturing long-term relationships by

$$p(\boldsymbol{x}_n|j, \boldsymbol{x}_{1:n-1})$$

Limitation of Conventional HMMs

Emission pdf of Conventional HMMs

$$p(\boldsymbol{x}_n|j)$$

$\Rightarrow$ conditional independence assumption

Conditional Emission pdf

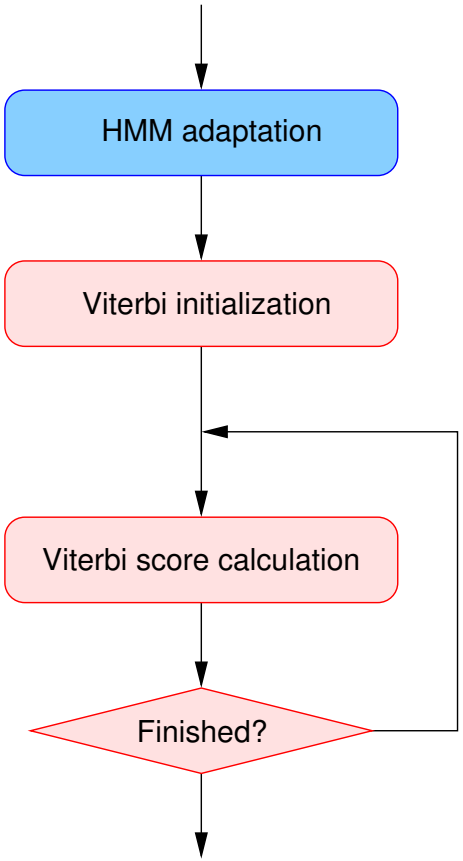
capturing long-term relationships by

$$p(\boldsymbol{x}_n|j, \boldsymbol{x}_{1:n-1})$$

Approximation by: **Context-aware Methods**:

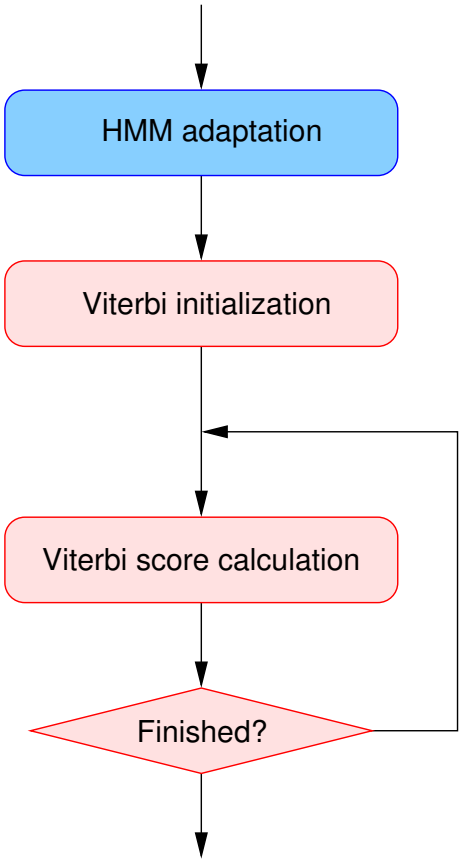
- Frame-wise HMM adaptation
- REMOS

Conventional Adaptation versus Context-Aware Methods

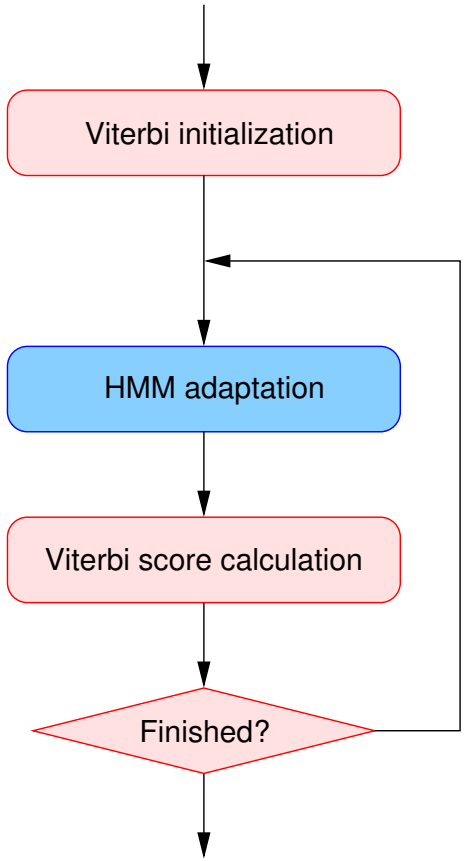


(a) Conventional HMM adaptation

Conventional Adaptation versus Context-Aware Methods

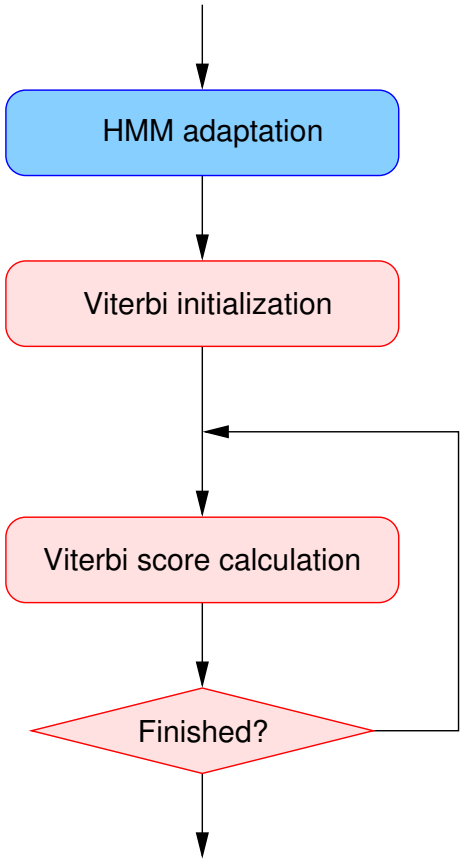


(a) Conventional HMM adaptation

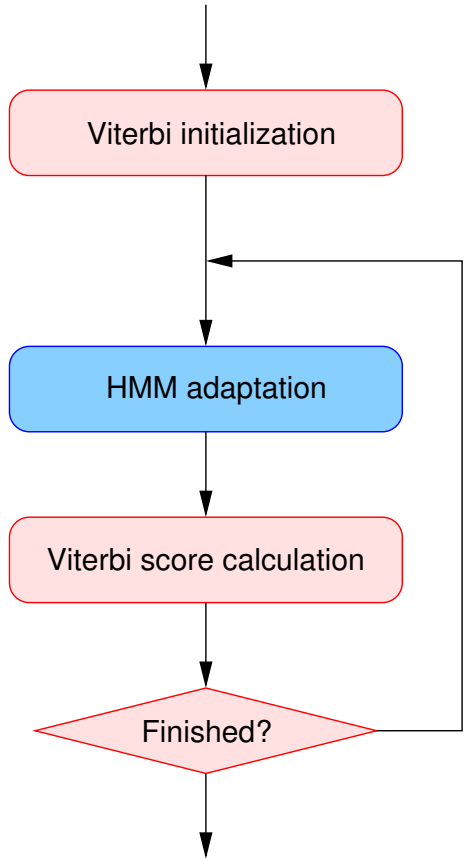


(b) Frame-wise adaptation

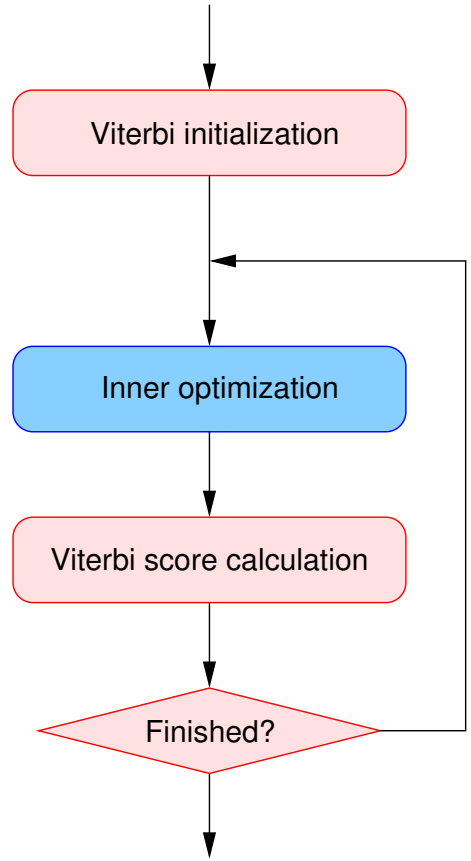
Conventional Adaptation versus Context-Aware Methods



(a) Conventional HMM adaptation



(b) Frame-wise adaptation



(c) REMOS

Frame-wise Adaptation

$$\boldsymbol{x}_n \approx \log(\exp(\boldsymbol{h}_0 + \boldsymbol{s}_n) + \exp(\boldsymbol{r}_n))$$

$$\boldsymbol{\mu}_{\boldsymbol{x}_n}(j) = \log(\exp(\boldsymbol{h}_0 + \boldsymbol{\mu}_{\boldsymbol{s}}(j)) + \exp(\boldsymbol{r}_n))$$

$\boldsymbol{r}_n$ late reverberation

j state index

Frame-wise Adaptation

$$\boldsymbol{x}_n \approx \log(\exp(\boldsymbol{h}_0 + \boldsymbol{s}_n) + \exp(\boldsymbol{r}_n))$$

$$\boldsymbol{\mu}_{\boldsymbol{x}_n}(j) = \log(\exp(\boldsymbol{h}_0 + \boldsymbol{\mu}_{\boldsymbol{s}}(j)) + \exp(\boldsymbol{r}_n))$$

$\boldsymbol{r}_n$ late reverberation

j state index

Autoregressive Modeling [Takiguchi 2006]

$$\boldsymbol{r}_n = \boldsymbol{a} + \boldsymbol{x}_{n-1}$$

$\boldsymbol{a}$ prediction coefficient

Frame-wise Adaptation

$$\begin{aligned} \boldsymbol{x}_n &\approx \log(\exp(\boldsymbol{h}_0 + \boldsymbol{s}_n) + \exp(\boldsymbol{r}_n)) \\ \boldsymbol{\mu}_{\boldsymbol{x}_n}(j) &= \log(\exp(\boldsymbol{h}_0 + \boldsymbol{\mu}_{\boldsymbol{s}}(j)) + \exp(\boldsymbol{r}_n)) \end{aligned}$$

$\boldsymbol{r}_n$ late reverberation
j state index

Autoregressive Modeling [Takiguchi 2006]

$$\boldsymbol{r}_n = \boldsymbol{a} + \boldsymbol{x}_{n-1}$$

$\boldsymbol{a}$ prediction coefficient

Moving-Average Modeling [Sehr 2011]

$$\boldsymbol{r}_n = \log\left(\sum_{\tau=1}^{T_H} \exp(\boldsymbol{\mu}_{\boldsymbol{h}_\tau} + \boldsymbol{s}_{n-\tau})\right)$$

Frame-wise Adaptation

Discussion

+ Overcomes conditional independence assumption
+ Accurate modeling of long-term relations

Frame-wise Adaptation

Discussion

+ Overcomes conditional independence assumption
+ Accurate modeling of long-term relations
– Increased computational complexity
– Increased effort for integration into ASR systems

Frame-wise Adaptation

Discussion

- \+ Overcomes conditional independence assumption
- \+ Accurate modeling of long-term relations
- – Increased computational complexity
- – Increased effort for integration into ASR systems
- ▸ Full potential not yet demonstrated

Frame-wise Adaptation

Discussion

- \+ Overcomes conditional independence assumption
- \+ Accurate modeling of long-term relations
- – Increased computational complexity
- – Increased effort for integration into ASR systems
- ▸ Full potential not yet demonstrated

Promising direction for future research

Further Model-based Approaches

[Couvreur 2001]	Reverberant training of several HMMs + model selection
[Sehr 2010b]	Training of reverberant HMMs on stereo data
[Gales 2011]	Extension of MLLR and VTS to reverberant environments

Part III: Robust ASR in Reverberant Environments

- Introduction
- Feature-based Approaches
- Model-based Approaches
- **Decoder-based Approaches**
- REMOS

Overview of Decoder-based Approaches

Key Idea

Modify the decoding algorithm to increase reverberation robustness

Overview of Decoder-based Approaches

Key Idea

Modify the decoding algorithm to increase reverberation robustness

Two Approaches

- **Missing feature techniques**
  - $\Rightarrow$ Distinguish between reliable and unreliable observations
  - $\Rightarrow$ Estimate or discard the unreliable parts

Overview of Decoder-based Approaches

Key Idea

Modify the decoding algorithm to increase reverberation robustness

Two Approaches

- **Missing feature techniques**
  - $\Rightarrow$ Distinguish between reliable and unreliable observations
  - $\Rightarrow$ Estimate or discard the unreliable parts
- **Uncertainty decoding**
  - $\Rightarrow$ Combined with signal or feature enhancement techniques
  - $\Rightarrow$ Exploit reliability information about enhanced data

Overview of Decoder-based Approaches

Key Idea

Modify the decoding algorithm to increase reverberation robustness

Two Approaches

- **Missing feature techniques**
  - $\Rightarrow$ Distinguish between reliable and unreliable observations
  - $\Rightarrow$ Estimate or discard the unreliable parts
- **Uncertainty decoding**
  - $\Rightarrow$ Combined with signal or feature enhancement techniques
  - $\Rightarrow$ Exploit reliability information about enhanced data

Decoder-based approaches bridge the gap between feature-based and model-based approaches

Missing Feature Techniques

For overviews see [Cooke 2001, Raj 2005, Kolossa 2011]

Key Ideas

- Partition the observations into reliable and missing components
- Use only the reliable components for recognition

Missing Feature Techniques

For overviews see [Cooke 2001, Raj 2005, Kolossa 2011]

Key Ideas

- Partition the observations into reliable and missing components
- Use only the reliable components for recognition

Main Steps

- Mask estimation: Mark observations as either reliable or missing
- Handle missing data appropriately

Missing Feature Techniques

For overviews see [Cooke 2001, Raj 2005, Kolossa 2011]

Key Ideas

- Partition the observations into reliable and missing components
- Use only the reliable components for recognition

Main Steps

- Mask estimation: Mark observations as either reliable or missing
- Handle missing data appropriately

How to handle missing data?

Missing Feature Techniques

For overviews see [Cooke 2001, Raj 2005, Kolossa 2011]

Key Ideas

- Partition the observations into reliable and missing components
- Use only the reliable components for recognition

Main Steps

- Mask estimation: Mark observations as either reliable or missing
- Handle missing data appropriately

How to handle missing data?

- **Marginalization**: Eliminate unreliable data by integration over corresponding dimensions

Missing Feature Techniques

For overviews see [Cooke 2001, Raj 2005, Kolossa 2011]

Key Ideas

- Partition the observations into reliable and missing components
- Use only the reliable components for recognition

Main Steps

- Mask estimation: Mark observations as either reliable or missing
- Handle missing data appropriately

How to handle missing data?

- **Marginalization**: Eliminate unreliable data by integration over corresponding dimensions
- **Bounded marginalization**: Exploit known bounds of the missing data for integration

Missing Feature Techniques

For overviews see [Cooke 2001, Raj 2005, Kolossa 2011]

Key Ideas

- Partition the observations into reliable and missing components
- Use only the reliable components for recognition

Main Steps

- Mask estimation: Mark observations as either reliable or missing
- Handle missing data appropriately

How to handle missing data?

- **Marginalization**: Eliminate unreliable data by integration over corresponding dimensions
- **Bounded marginalization**: Exploit known bounds of the missing data for integration
- **Data imputation**: Determine state-dependent estimates for the unreliable data, given the reliable data

Missing Feature Techniques for Reverberation Robustness

[Palomäki 2004]

- Modulation filtering for the mask estimation
- Bounded marginalization for handling missing data

Missing Feature Techniques for Reverberation Robustness

[Palomäki 2004]

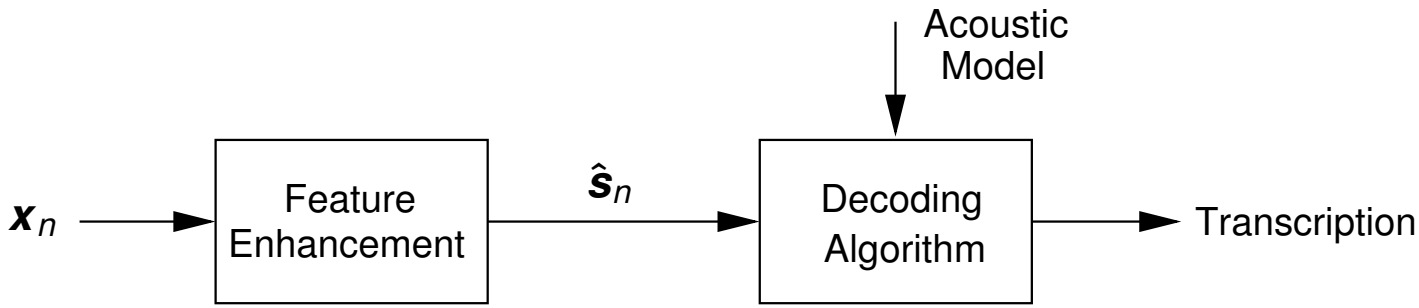
- Modulation filtering for the mask estimation
- Bounded marginalization for handling missing data

[Gemmeke 2011]

- Oracle masks based on clean and reverberant features
- Semi-Oracle masks based on clean features and estimated RIRs
- Gaussian-dependent bounded imputation

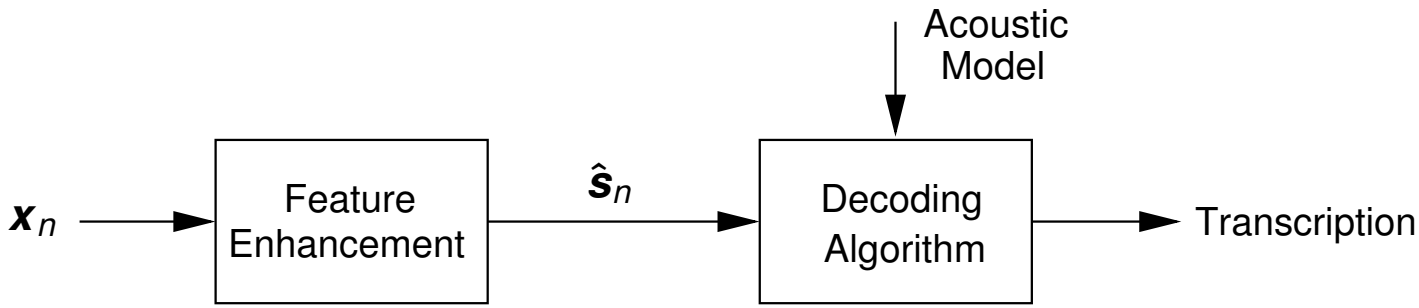
Uncertainty Decoding

Conventional Feature Enhancement Methods



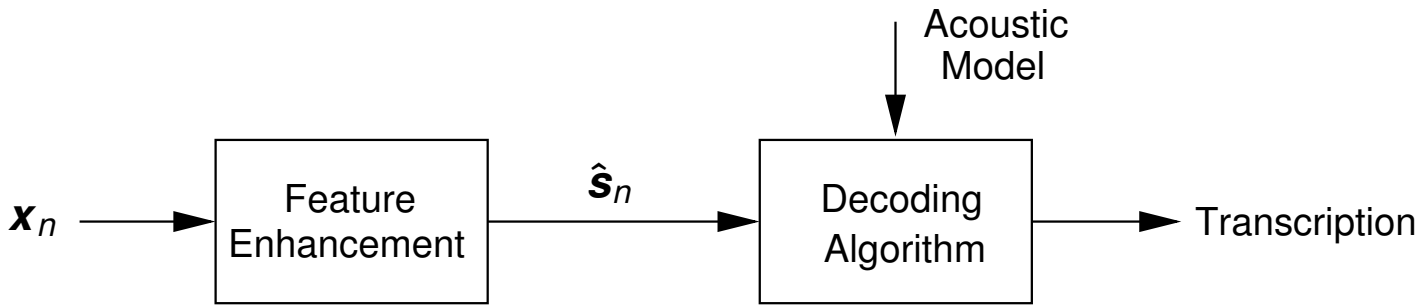
Uncertainty Decoding

Conventional Feature Enhancement Methods



- Use only point estimate $\hat{\boldsymbol{s}}_n$ of clean features

Conventional Feature Enhancement Methods



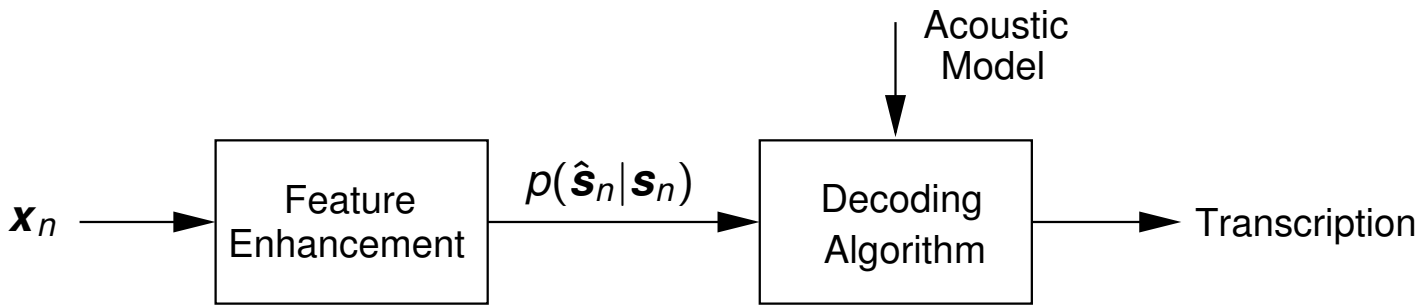
- Use only point estimate $\hat{\boldsymbol{s}}_n$ of clean features
- Contribution of each Gaussian component m:

$$p(\hat{\boldsymbol{s}}_n|m) = \mathcal{N}(\hat{\boldsymbol{s}}_n; \boldsymbol{\mu}_{\boldsymbol{s}}^{(m)}, \boldsymbol{\Sigma}_{\boldsymbol{s}}^{(m)})$$

Uncertainty Decoding

[Droppo 2002, Deng 2005, Liao 2008, Haeb-Umbach 2011]

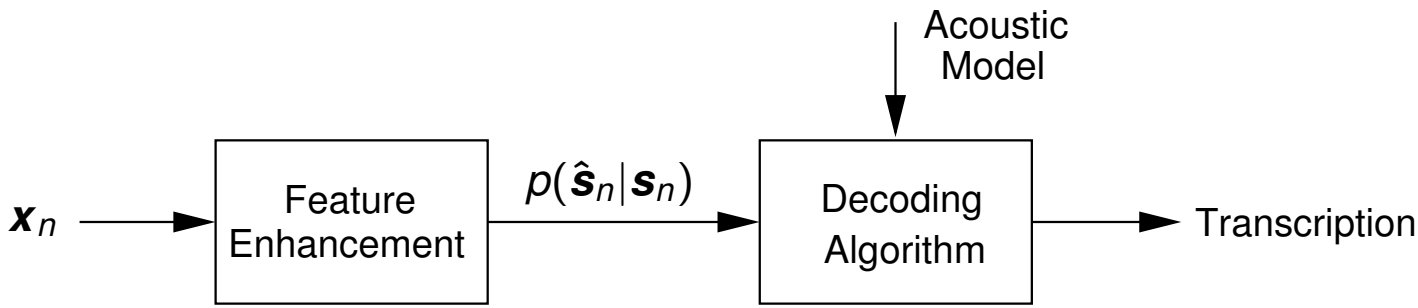
Feature Enhancement Combined with Uncertainty Decoding



Uncertainty Decoding

[Droppo 2002, Deng 2005, Liao 2008, Haeb-Umbach 2011]

Feature Enhancement Combined with Uncertainty Decoding

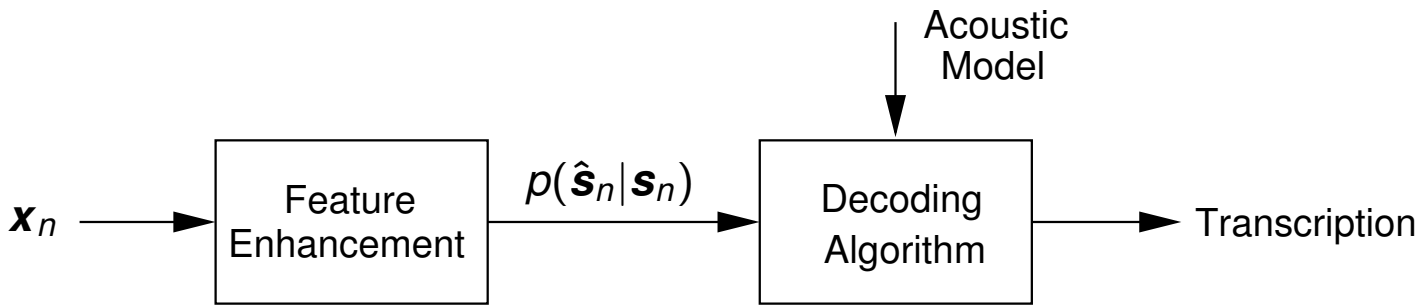


- Signal/feature enhancement inevitably introduces distortions

Uncertainty Decoding

[Droppo 2002, Deng 2005, Liao 2008, Haeb-Umbach 2011]

Feature Enhancement Combined with Uncertainty Decoding



- Signal/feature enhancement inevitably introduces distortions
- Use reliability information in addition to point estimate
  $\Rightarrow$ Use $p(\hat{\boldsymbol{s}}_n|\boldsymbol{s}_n)$ instead of $\hat{\boldsymbol{s}}_n$

Uncertainty Decoding

[Droppo 2002, Deng 2005, Liao 2008, Haeb-Umbach 2011]

Mismatch Model

$$\begin{aligned} \hat{\boldsymbol{s}}_n &= \boldsymbol{s}_n + \boldsymbol{b}_n \\ p(\boldsymbol{b}_n) &= \mathcal{N}(\boldsymbol{b}_n; \boldsymbol{0}, \Sigma_{\boldsymbol{b}_n}) \\ p(\hat{\boldsymbol{s}}_n|\boldsymbol{s}_n, m) &\approx p(\hat{\boldsymbol{s}}_n|\boldsymbol{s}_n) = p(\boldsymbol{b}_n) \end{aligned}$$

Uncertainty Decoding

[Droppo 2002, Deng 2005, Liao 2008, Haeb-Umbach 2011]

Mismatch Model

$$\begin{aligned} \hat{\boldsymbol{s}}_n &= \boldsymbol{s}_n + \boldsymbol{b}_n \\ p(\boldsymbol{b}_n) &= \mathcal{N}(\boldsymbol{b}_n; \boldsymbol{0}, \Sigma_{\boldsymbol{b}_n}) \\ p(\hat{\boldsymbol{s}}_n|\boldsymbol{s}_n, m) &\approx p(\hat{\boldsymbol{s}}_n|\boldsymbol{s}_n) = p(\boldsymbol{b}_n) \end{aligned}$$

Contribution of Gaussian Component m

$$p(\hat{\boldsymbol{s}}_n|m) = \int p(\hat{\boldsymbol{s}}_n, \boldsymbol{s}_n|m)\, \mathrm{d}\boldsymbol{s}_n = \int p(\hat{\boldsymbol{s}}_n|\boldsymbol{s}_n, m)\, p(\boldsymbol{s}_n|m)\, \mathrm{d}\boldsymbol{s}_n$$

Uncertainty Decoding

[Droppo 2002, Deng 2005, Liao 2008, Haeb-Umbach 2011]

Mismatch Model

$$\begin{aligned} \hat{\boldsymbol{s}}_n &= \boldsymbol{s}_n + \boldsymbol{b}_n \\ p(\boldsymbol{b}_n) &= \mathcal{N}(\boldsymbol{b}_n; \boldsymbol{0}, \Sigma_{\boldsymbol{b}_n}) \\ p(\hat{\boldsymbol{s}}_n|\boldsymbol{s}_n, m) &\approx p(\hat{\boldsymbol{s}}_n|\boldsymbol{s}_n) = p(\boldsymbol{b}_n) \end{aligned}$$

Contribution of Gaussian Component *m*

$$\begin{aligned} p(\hat{\boldsymbol{s}}_n|m) &= \int p(\hat{\boldsymbol{s}}_n, \boldsymbol{s}_n|m)\, \mathrm{d}\boldsymbol{s}_n = \int p(\hat{\boldsymbol{s}}_n|\boldsymbol{s}_n, m)\, p(\boldsymbol{s}_n|m)\, \mathrm{d}\boldsymbol{s}_n \\ &\approx \mathcal{N}(\hat{\boldsymbol{s}}_n; \boldsymbol{\mu}_{\boldsymbol{s}}^{(m)}, \Sigma_{\boldsymbol{s}}^{(m)} + \Sigma_{\boldsymbol{b}_n}) \end{aligned}$$

Uncertainty Decoding

[Droppo 2002, Deng 2005, Liao 2008, Haeb-Umbach 2011]

Mismatch Model

$$\begin{aligned} \hat{\boldsymbol{s}}_n &= \boldsymbol{s}_n + \boldsymbol{b}_n \\ p(\boldsymbol{b}_n) &= \mathcal{N}(\boldsymbol{b}_n; \boldsymbol{0}, \Sigma_{\boldsymbol{b}_n}) \\ p(\hat{\boldsymbol{s}}_n|\boldsymbol{s}_n, m) &\approx p(\hat{\boldsymbol{s}}_n|\boldsymbol{s}_n) = p(\boldsymbol{b}_n) \end{aligned}$$

Contribution of Gaussian Component *m*

$$\begin{aligned} p(\hat{\boldsymbol{s}}_n|m) &= \int p(\hat{\boldsymbol{s}}_n, \boldsymbol{s}_n|m)\, \mathrm{d}\boldsymbol{s}_n = \int p(\hat{\boldsymbol{s}}_n|\boldsymbol{s}_n, m)\, p(\boldsymbol{s}_n|m)\, \mathrm{d}\boldsymbol{s}_n \\ &\approx \mathcal{N}(\hat{\boldsymbol{s}}_n; \boldsymbol{\mu}_{\boldsymbol{s}}^{(m)}, \Sigma_{\boldsymbol{s}}^{(m)} + \Sigma_{\boldsymbol{b}_n}) \end{aligned}$$

- Unreliable features $\Rightarrow$ large $\Sigma_{\boldsymbol{b}_n}$ $\Rightarrow$ little effect on Viterbi score

Uncertainty Decoding

[Droppo 2002, Deng 2005, Liao 2008, Haeb-Umbach 2011]

Mismatch Model

$$\begin{aligned} \hat{\boldsymbol{s}}_n &= \boldsymbol{s}_n + \boldsymbol{b}_n \\ p(\boldsymbol{b}_n) &= \mathcal{N}(\boldsymbol{b}_n; \boldsymbol{0}, \Sigma_{\boldsymbol{b}_n}) \\ p(\hat{\boldsymbol{s}}_n|\boldsymbol{s}_n, m) &\approx p(\hat{\boldsymbol{s}}_n|\boldsymbol{s}_n) = p(\boldsymbol{b}_n) \end{aligned}$$

Contribution of Gaussian Component *m*

$$\begin{aligned} p(\hat{\boldsymbol{s}}_n|m) &= \int p(\hat{\boldsymbol{s}}_n, \boldsymbol{s}_n|m)\, \mathrm{d}\boldsymbol{s}_n = \int p(\hat{\boldsymbol{s}}_n|\boldsymbol{s}_n, m)\, p(\boldsymbol{s}_n|m)\, \mathrm{d}\boldsymbol{s}_n \\ &\approx \mathcal{N}(\hat{\boldsymbol{s}}_n; \boldsymbol{\mu}_{\boldsymbol{s}}^{(m)}, \Sigma_{\boldsymbol{s}}^{(m)} + \Sigma_{\boldsymbol{b}_n}) \end{aligned}$$

- Unreliable features $\Rightarrow$ large $\Sigma_{\boldsymbol{b}_n}$ $\Rightarrow$ little effect on Viterbi score
- **Main challenge**: Estimation of time-variant feature cov. $\Sigma_{\boldsymbol{b}_n}$

Uncertainty Decoding for Reverberation-Robust ASR

[Delcroix 2009, Delcroix 2011a]

Key Idea

- Strong reverberation
  $\Rightarrow$ Large effect of speech enhancement
  $\Rightarrow$ Large mismatch between clean and enhanced features

Uncertainty Decoding for Reverberation-Robust ASR

[Delcroix 2009, Delcroix 2011a]

Key Idea

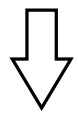
- Strong reverberation
  $\Rightarrow$ Large effect of speech enhancement
  $\Rightarrow$ Large mismatch between clean and enhanced features
- Effect of speech enhancement captured by $\hat{\boldsymbol{b}}_n = \boldsymbol{x}_n - \hat{\boldsymbol{s}}_n$
  $\Rightarrow$ Mismatch covariance assumed proportional to difference between observed and enhanced features

Uncertainty Decoding for Reverberation-Robust ASR

[Delcroix 2009, Delcroix 2011a]

Key Idea

- Strong reverberation
  $\Rightarrow$ Large effect of speech enhancement
  $\Rightarrow$ Large mismatch between clean and enhanced features
- Effect of speech enhancement captured by $\hat{\boldsymbol{b}}_n = \boldsymbol{x}_n - \hat{\boldsymbol{s}}_n$
  $\Rightarrow$ Mismatch covariance assumed proportional to difference between observed and enhanced features



Model elements of time-variant diagonal mismatch cov. matrix $\boldsymbol{\Sigma}_{\boldsymbol{b}_n}$ as

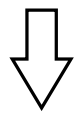
$$(\boldsymbol{\Sigma}_{\boldsymbol{b}_n})_{ii} \quad = \quad \alpha_i \, \hat{b}_{n,i}^2$$

Uncertainty Decoding for Reverberation-Robust ASR

[Delcroix 2009, Delcroix 2011a]

Key Idea

- Strong reverberation
  $\Rightarrow$ Large effect of speech enhancement
  $\Rightarrow$ Large mismatch between clean and enhanced features
- Effect of speech enhancement captured by $\hat{\boldsymbol{b}}_n = \boldsymbol{x}_n - \hat{\boldsymbol{s}}_n$
  $\Rightarrow$ Mismatch covariance assumed proportional to difference between observed and enhanced features



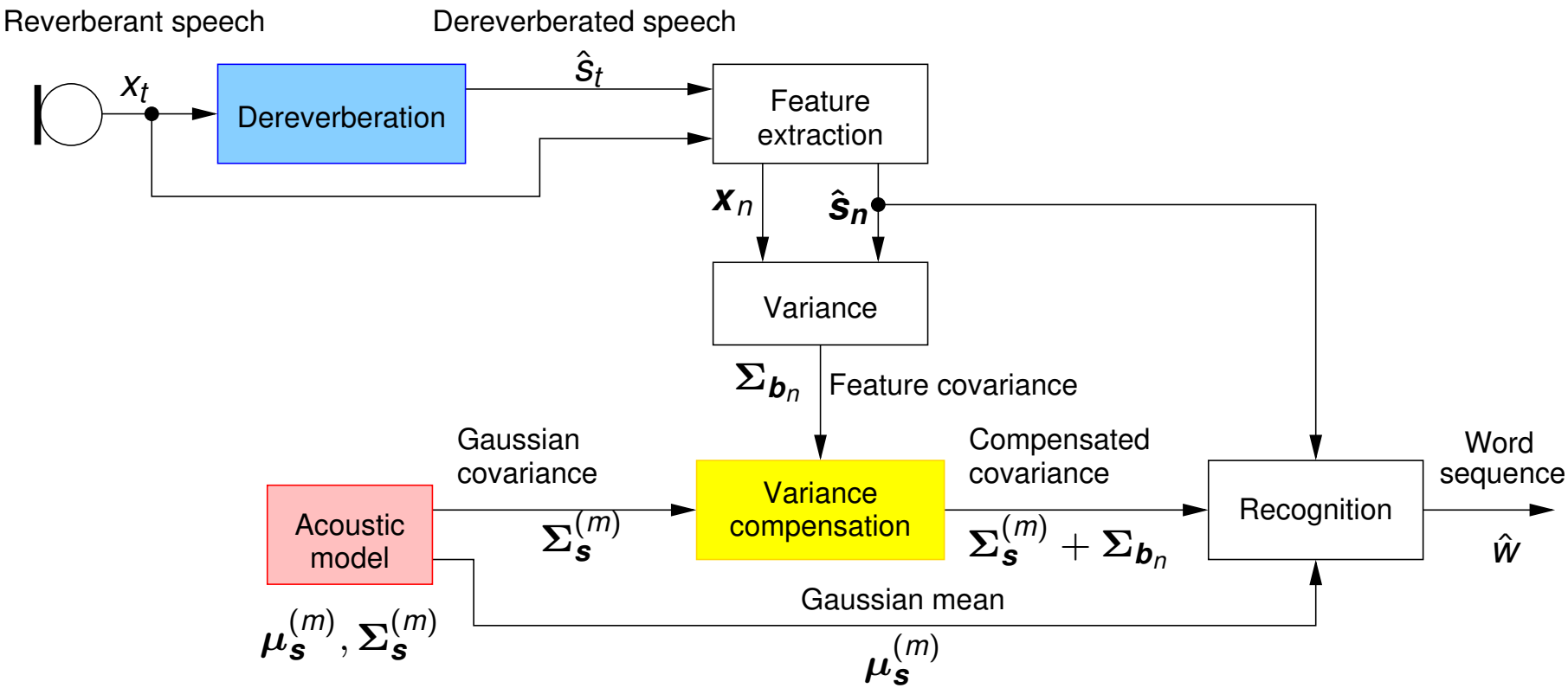
Model elements of time-variant diagonal mismatch cov. matrix $\boldsymbol{\Sigma}_{\boldsymbol{b}_n}$ as

$$(\boldsymbol{\Sigma}_{\boldsymbol{b}_n})_{ii} \quad = \quad \alpha_i \, \hat{b}_{n,i}^2$$

- α is estimated by EM algorithm using adaptation data

Uncertainty Decoding for Reverberation-Robust ASR

[Delcroix 2009, Delcroix 2011a]



Uncertainty Decoding for Reverberation-Robust ASR

[Delcroix 2009, Delcroix 2011a]

Discussion

- Accounting for time-variant covariance matrix $\Sigma_{\boldsymbol{b}_n}$ increases computational complexity

Uncertainty Decoding for Reverberation-Robust ASR

[Delcroix 2009, Delcroix 2011a]

Discussion

- – Accounting for time-variant covariance matrix $\Sigma_{\boldsymbol{b}_n}$ increases computational complexity
- + Can be combined with static variance compensation and mean adaptation by MLLR
- + Independent of enhancement algorithm $\Rightarrow$ Highly flexible
- + Has been used successfully also for non-stationary interferences [Delcroix 2011b]

Uncertainty Decoding for Reverberation-Robust ASR

[Delcroix 2009, Delcroix 2011a]

Discussion

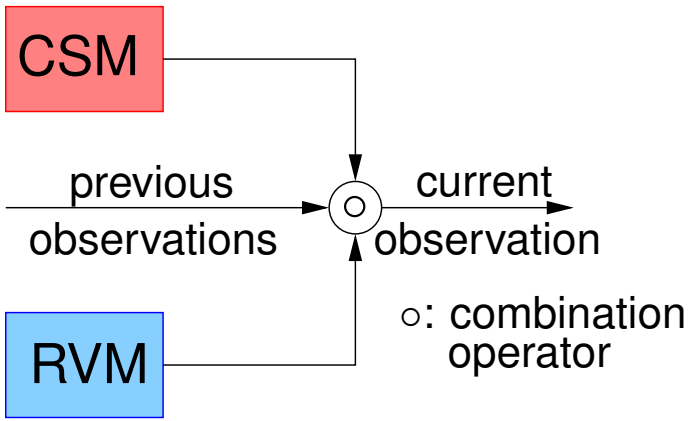
- – Accounting for time-variant covariance matrix $\Sigma_{\boldsymbol{b}_n}$ increases computational complexity
- \+ Can be combined with static variance compensation and mean adaptation by MLLR
- \+ Independent of enhancement algorithm $\Rightarrow$ Highly flexible
- \+ Has been used successfully also for non-stationary interferences [Delcroix 2011b]

Promising approach for interconnection of signal/feature-based methods and ASR systems

Part III: Robust ASR in Reverberant Environments

- Introduction
- Feature-based Approaches
- Model-based Approaches
- Decoder-based Approaches
- REMOS

Online Model Combination



CSM: clean-speech model
$\Rightarrow$ HMM network

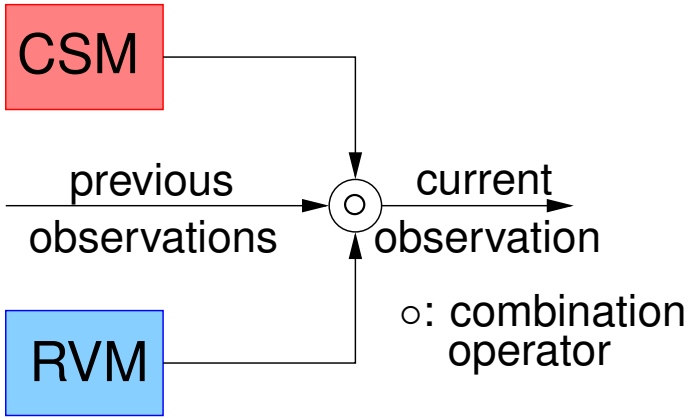
RVM: reverberation model

combination of CSM and RVM:

$\Rightarrow$ context-aware acoustic model

REMOS: REverberation MOdeling for Speech Recognition

Online Model Combination



CSM: clean-speech model
$\Rightarrow$ HMM network

RVM: reverberation model

combination of CSM and RVM:

$\Rightarrow$ context-aware acoustic model

Advantages

CSM and RVM are trained independently

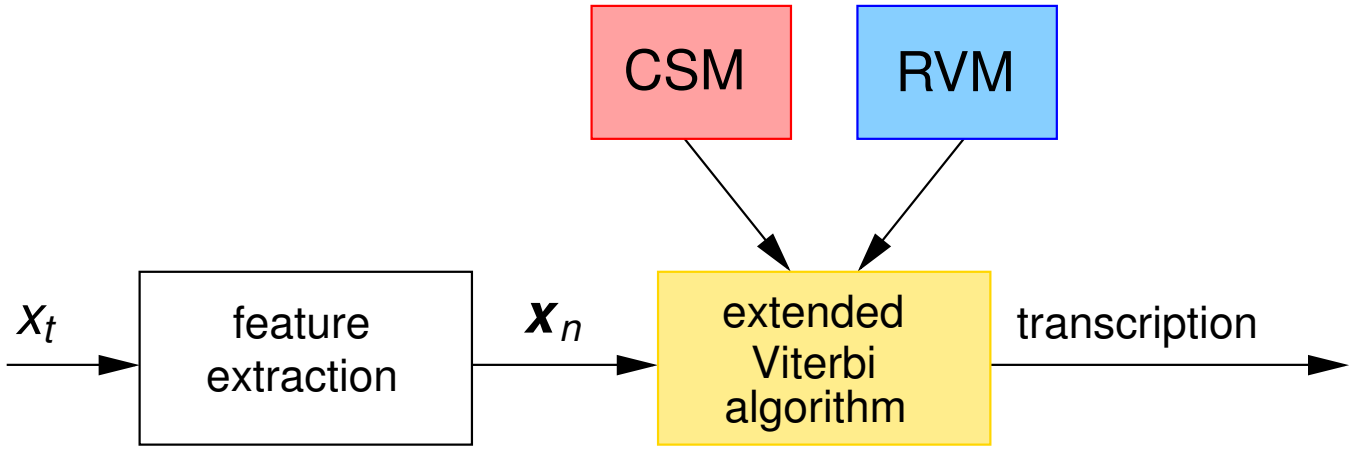
- changing environment: adjust only RVM
- changing task: adjust only CSM

$\Downarrow$

high degree of flexibility

[Sehr 2010c]

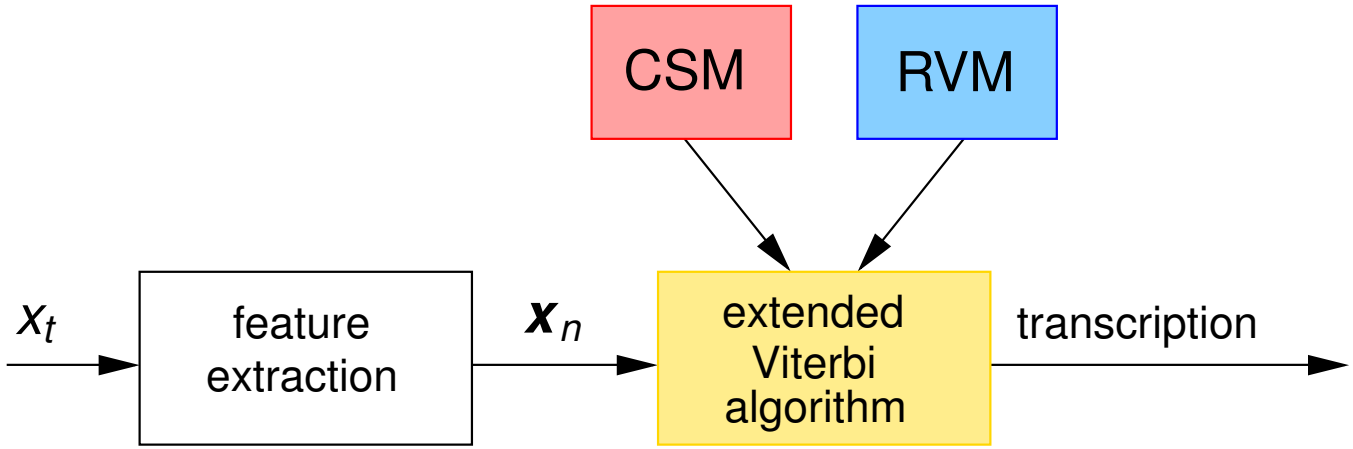
REMOS Decoding [Sehr 2010c]



- **Extended Viterbi Algorithm:**

  finds most likely path through CSM

REMOS Decoding [Sehr 2010c]



- **Extended Viterbi Algorithm:**

  finds most likely path through CSM

- **Inner Optimization:**
  - accounts for RVM and previous observations
  - determines most likely contributions of CSM and RVM to current observation

REMOS [Sehr 2010c]

Online combination of model outputs from clean-speech HMM and reverberation model capturing long-term relations:

REMOS [Sehr 2010c]

Online combination of model outputs from clean-speech HMM and reverberation model capturing long-term relations:

Combination Operator

$$\begin{aligned} \boldsymbol{x}_n &= \boldsymbol{f}(\boldsymbol{s}_n, \boldsymbol{s}_{n-T_H:n-1}, \boldsymbol{h}_n, \boldsymbol{a}_n) \\ &= \log(\exp(\boldsymbol{h}_n + \boldsymbol{s}_n) + \exp(\boldsymbol{r}_n + \boldsymbol{a}_n)) \end{aligned}$$

- $\boldsymbol{r}_n$: logmelspec late reverberation estimate
- $\boldsymbol{a}_n$: captures approximation error of $\boldsymbol{r}_n$
- $\boldsymbol{h}_n$: logmelspec representation of direct sound component of RIR

REMOS [Sehr 2010c]

Online combination of model outputs from clean-speech HMM and reverberation model capturing long-term relations:

Combination Operator

$$\begin{aligned} \boldsymbol{x}_n &= \boldsymbol{f}(\boldsymbol{s}_n, \boldsymbol{s}_{n-T_H:n-1}, \boldsymbol{h}_n, \boldsymbol{a}_n) \\ &= \log(\exp(\boldsymbol{h}_n + \boldsymbol{s}_n) + \exp(\boldsymbol{r}_n + \boldsymbol{a}_n)) \end{aligned}$$

Late Reverberation Estimate

$$\boldsymbol{r}_n = \log\left(\sum_{\tau=1}^{T_H} \exp(\boldsymbol{\mu}_\tau^H + \boldsymbol{s}_{n-\tau})\right)$$

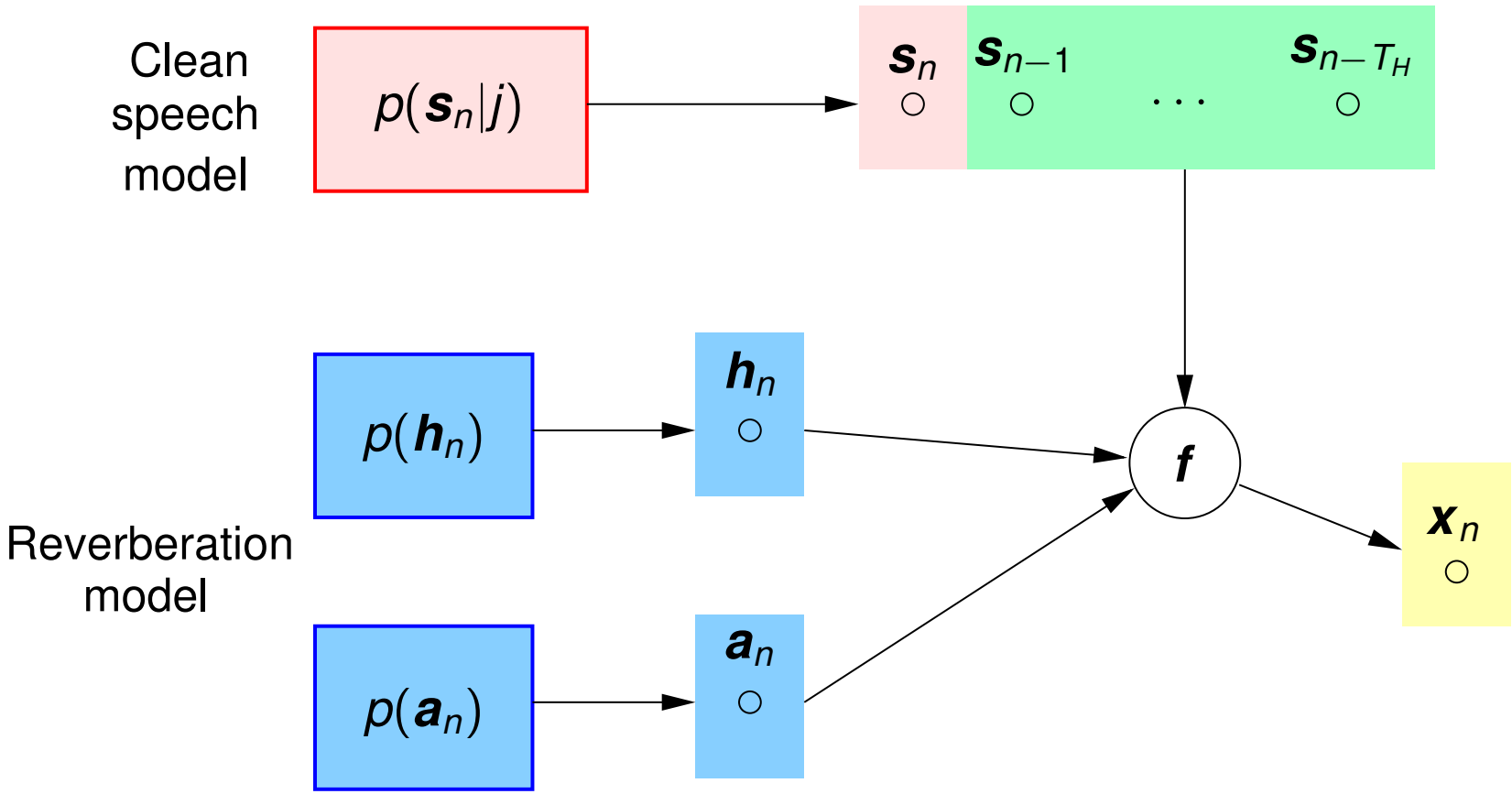
$\boldsymbol{r}_n$: logmelspec late reverberation estimate

$\boldsymbol{a}_n$: captures approximation error of $\boldsymbol{r}_n$

$\boldsymbol{h}_n$: logmelspec representation of direct sound component of RIR

$\boldsymbol{\mu}_{1:T_H}^H$: mean vectors of logmelspec representation for late reverberation

Illustration of Generative Model



REMOS [Sehr 2010c]

Conditional emission pdf is decomposed into reverberation model and clean HMM:

$$p(\boldsymbol{x}_n|j, \boldsymbol{x}_{1:n-1}) = \int p(\boldsymbol{x}_n|\boldsymbol{s}_n, \boldsymbol{x}_{1:n-1})\ p(\boldsymbol{s}_n|j)\ \mathrm{d}\boldsymbol{s}_n,$$

REMOS [Sehr 2010c]

Conditional emission pdf is decomposed into reverberation model and clean HMM:

$$p(\boldsymbol{x}_n|j, \boldsymbol{x}_{1:n-1}) = \int p(\boldsymbol{x}_n|\boldsymbol{s}_n, \boldsymbol{x}_{1:n-1})\ p(\boldsymbol{s}_n|j)\ \mathrm{d}\boldsymbol{s}_n,$$

Reverberation Model:

$$p(\boldsymbol{x}_n|\boldsymbol{s}_n, \boldsymbol{x}_{1:n-1}) = \iint p(\boldsymbol{h}_n)p(\boldsymbol{a}_n)\ \delta(\boldsymbol{x}_n - \boldsymbol{f}(\boldsymbol{s}_n, \boldsymbol{s}_{n-T_H:n-1}, \boldsymbol{h}_n, \boldsymbol{a}_n))\ \mathrm{d}\boldsymbol{h}_n\, \mathrm{d}\boldsymbol{a}_n$$

REMOS [Sehr 2010c]

Approximation of Conditional Emission pdf:
by maximum values of integrand

$$p(\boldsymbol{x}_n|j, \boldsymbol{x}_{1:n-1}) \quad \approx \quad p(\hat{\boldsymbol{h}}_n)\ p(\hat{\boldsymbol{a}}_n)\ p(\hat{\boldsymbol{s}}_n|j)$$

REMOS [Sehr 2010c]

Approximation of Conditional Emission pdf:
by maximum values of integrand

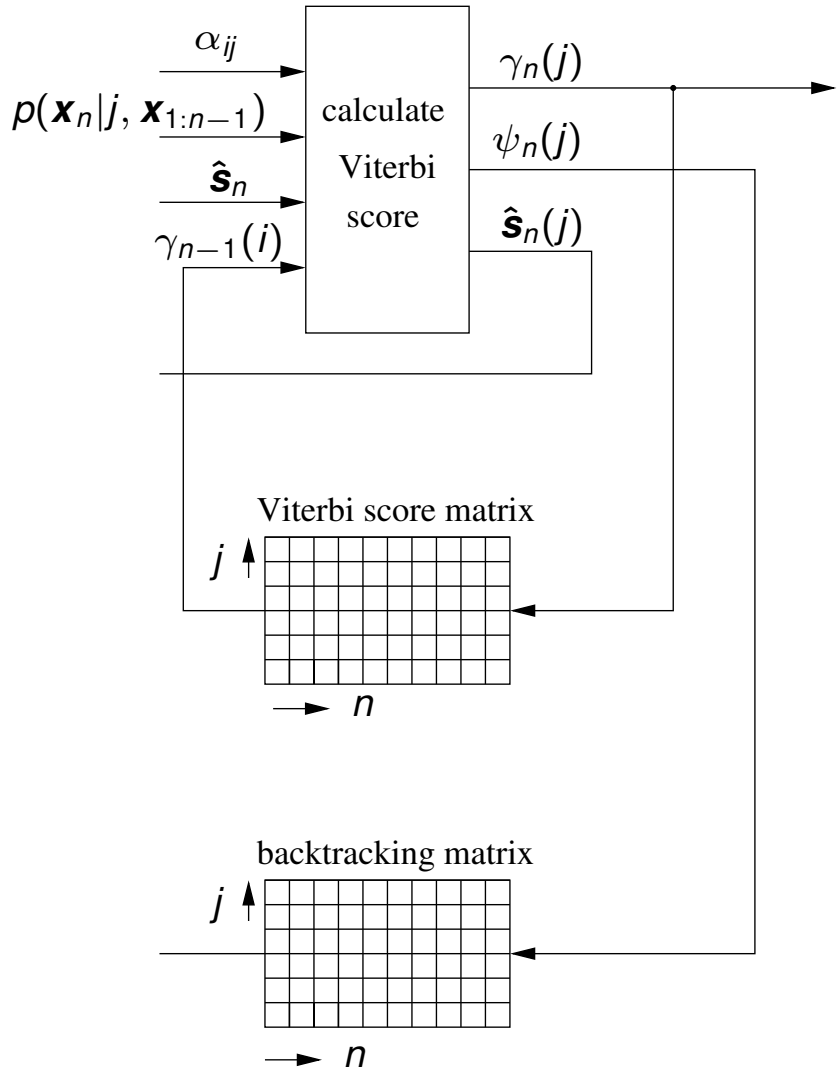
$$p(\boldsymbol{x}_n|j, \boldsymbol{x}_{1:n-1}) \approx p(\hat{\boldsymbol{h}}_n)\ p(\hat{\boldsymbol{a}}_n)\ p(\hat{\boldsymbol{s}}_n|j)$$

maximum values $\hat{\boldsymbol{h}}_n, \hat{\boldsymbol{a}}_n, \hat{\boldsymbol{s}}_n$ determined by **inner optimization**

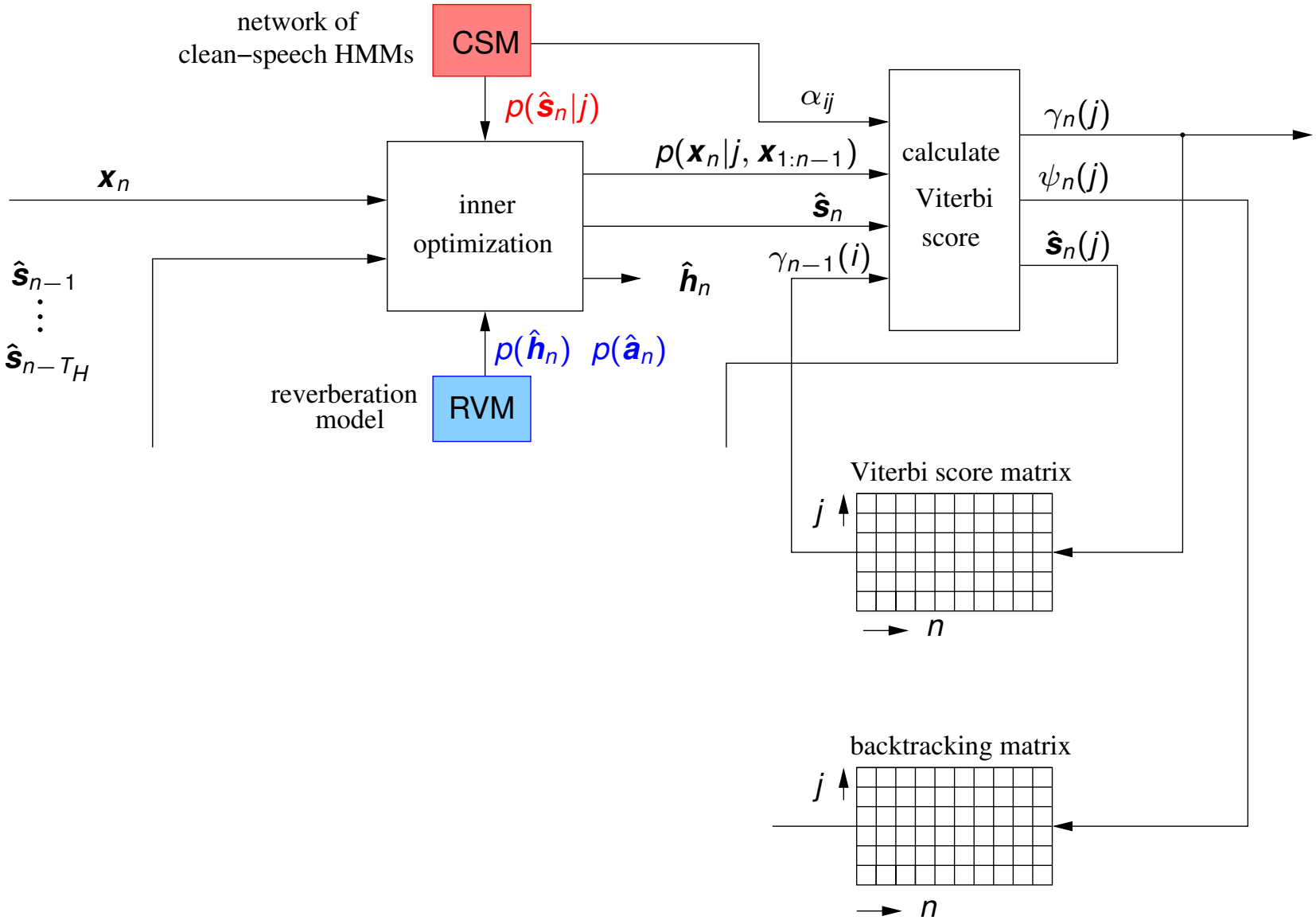
$$(\hat{\boldsymbol{h}}_n, \hat{\boldsymbol{a}}_n, \hat{\boldsymbol{s}}_n) = \underset{(\boldsymbol{h}_n, \boldsymbol{a}_n, \boldsymbol{s}_n)}{\operatorname{argmax}} p(\boldsymbol{h}_n)p(\boldsymbol{a}_n)p(\boldsymbol{s}_n|j)$$

$$\text{subject to} \quad \boldsymbol{x}_n = \boldsymbol{f}(\boldsymbol{s}_n, \boldsymbol{s}_{n-T_H:n-1}, \boldsymbol{h}_n, \boldsymbol{a}_n)$$

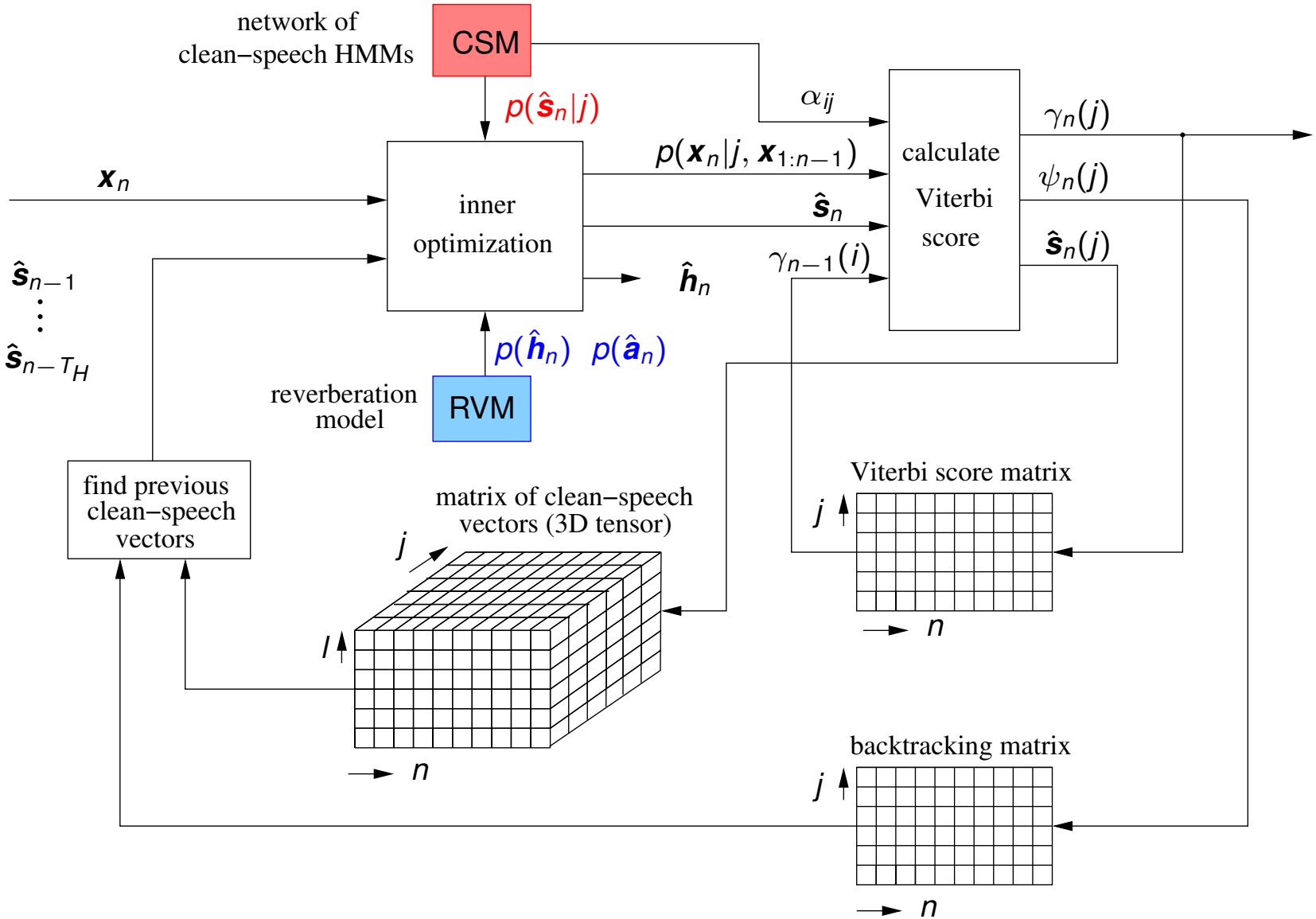
Detailed Illustration of REMOS Decoding



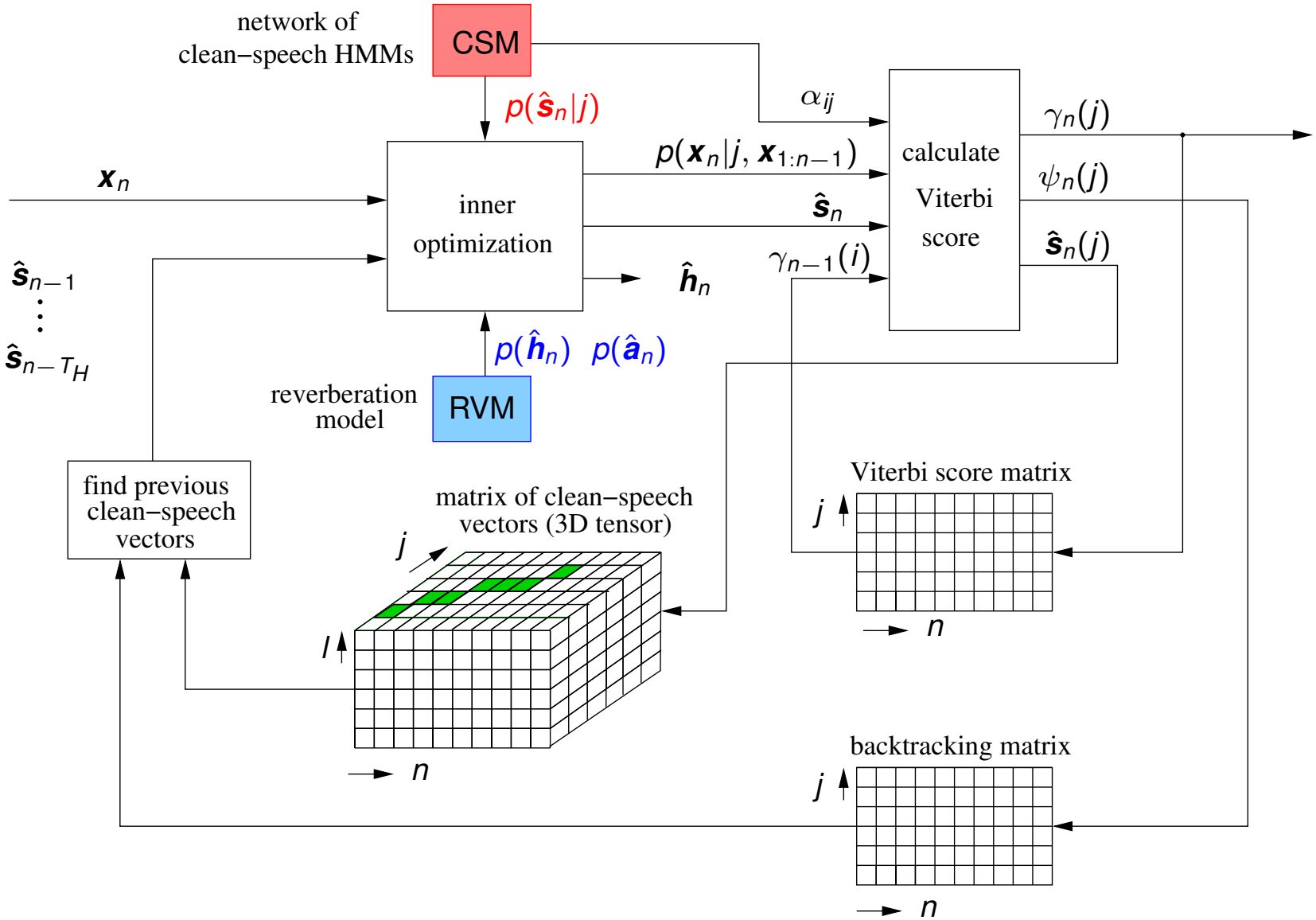
Detailed Illustration of REMOS Decoding



Detailed Illustration of REMOS Decoding

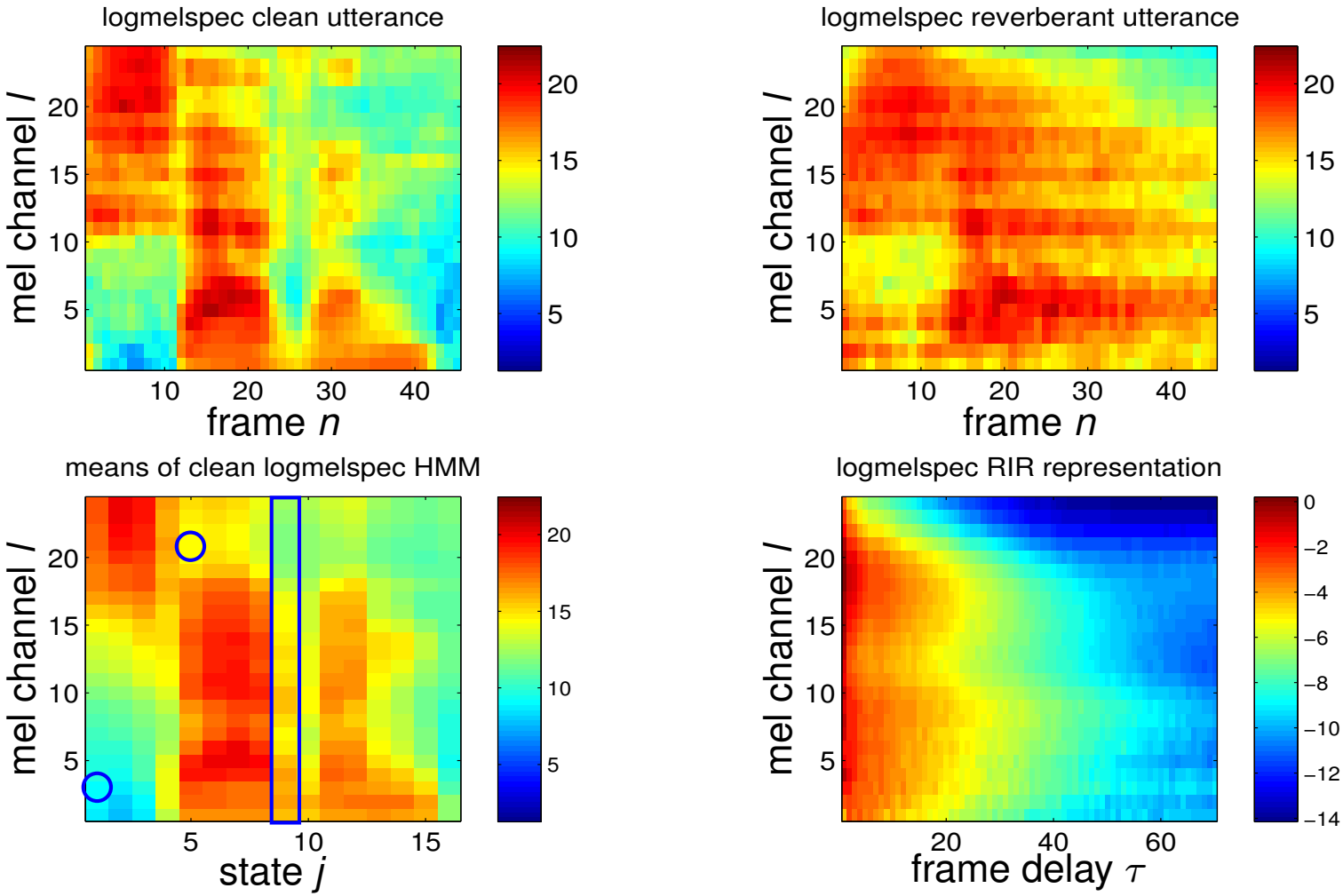


Detailed Illustration of REMOS Decoding



Modeling Accuracy of REMOS

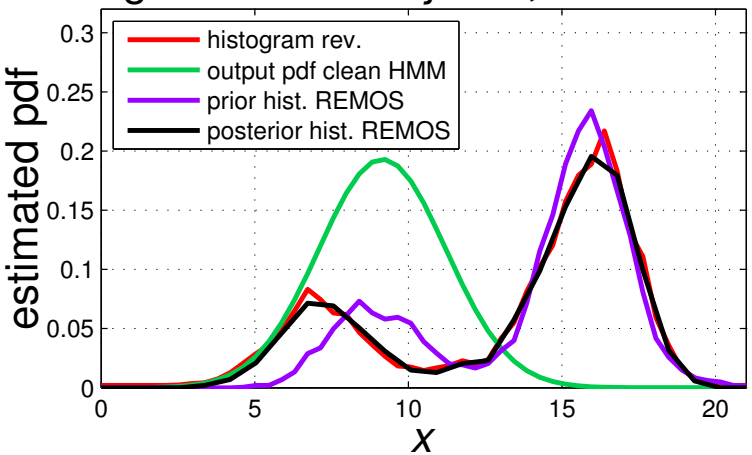
Example: digit “seven”



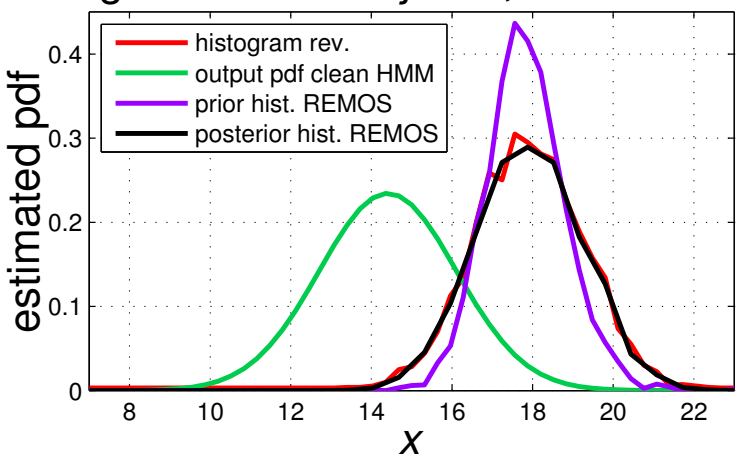
Modeling Accuracy of REMOS

Histograms

histograms for state $j = 1$, channel $l = 3$



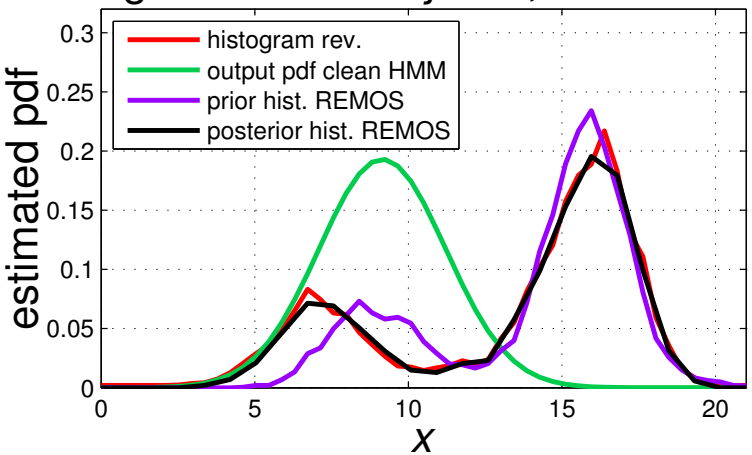
histograms for state $j = 5$, channel $l = 21$



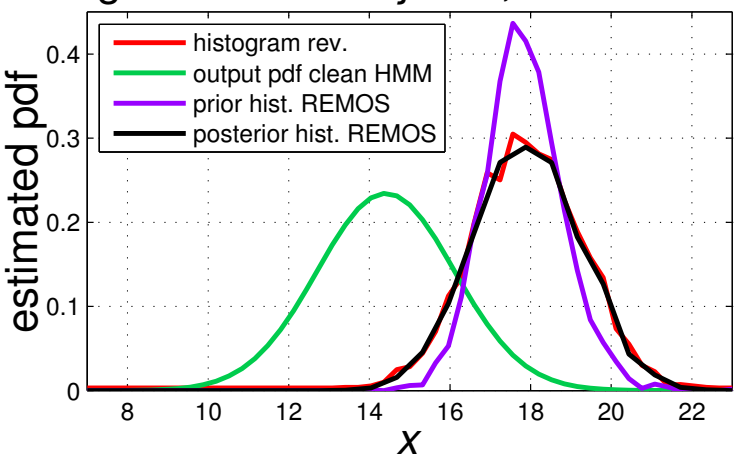
Modeling Accuracy of REMOS

Histograms

histograms for state $j = 1$, channel $l = 3$

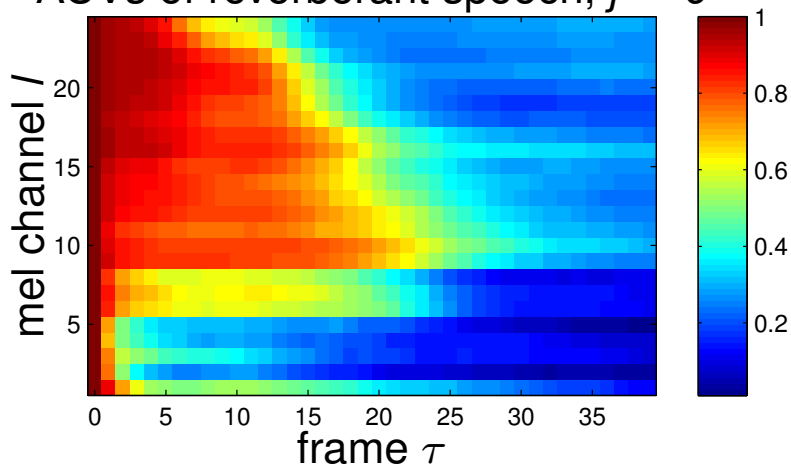


histograms for state $j = 5$, channel $l = 21$

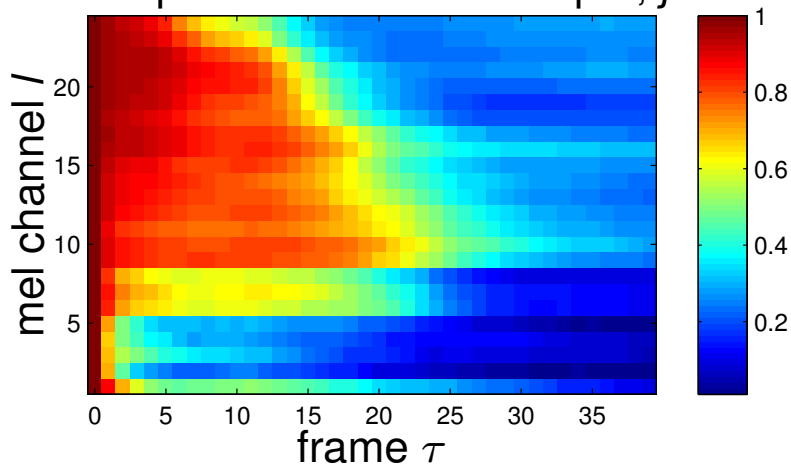


Auto-CoVariances (ACVs)

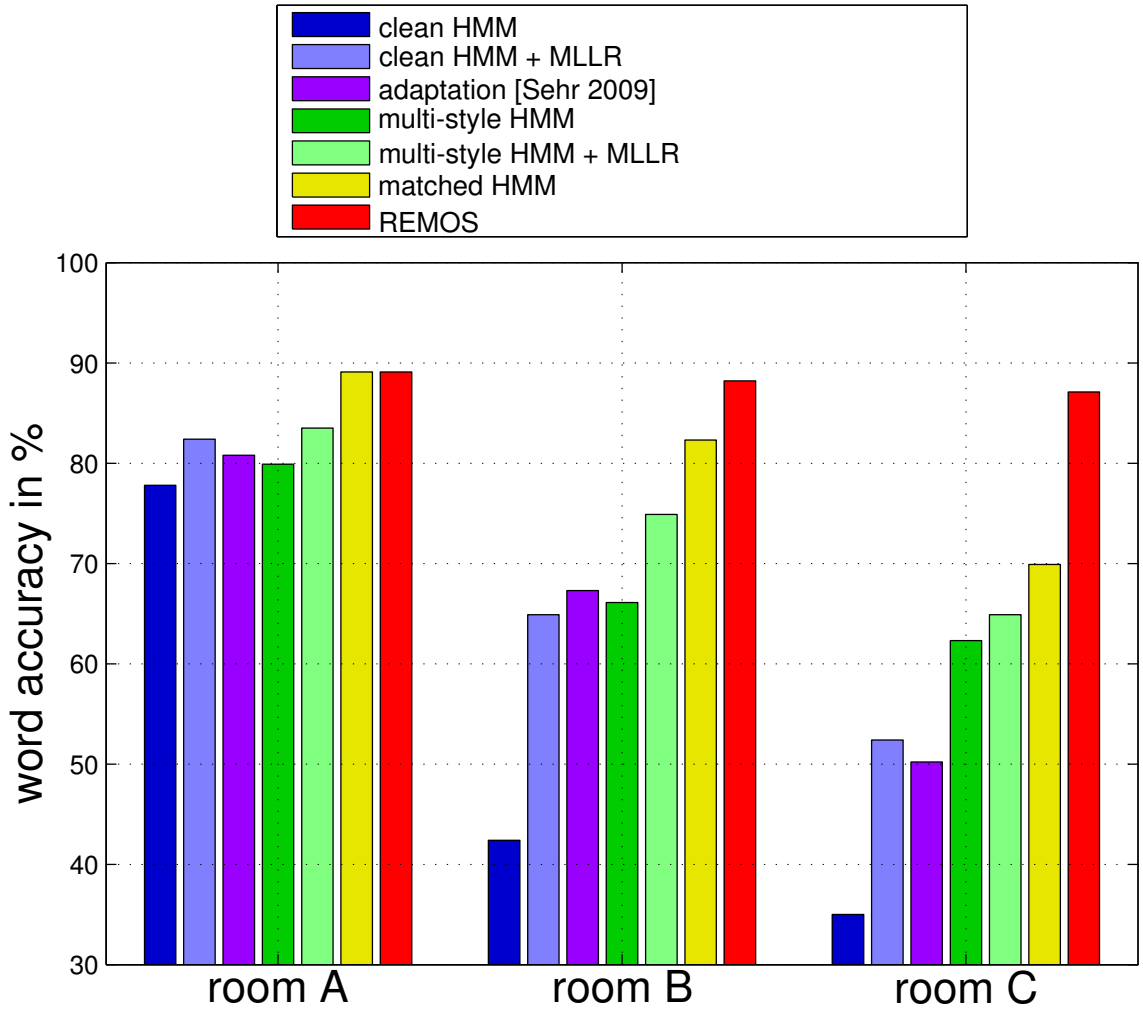
ACVs of reverberant speech, $j = 9$



ACVs of posterior REMOS output, $j = 9$



Recognition Results [Sehr 2010c]



Setup

- **Task:** Connected digits (TI digits)
- **Features:** Logmelspec coefficients
- **Recognizer:** Word-level HMMs, 16 states/digit, 1 Gaussian/state
- **Rooms:**

	T_{60}	*DRR*
A:	300 ms	4.0dB
B:	700 ms	−4.0dB
C:	900 ms	−4.0dB

REMOS [Sehr 2010c]

Discussion

+ Approach tailored to reverberant feature vector sequences
+ Long-term relations explicitely captured by reverberation model
+ Reverberation exploited for discrimination
+ Very promising results in logmelspec domain

REMOS [Sehr 2010c]

Discussion

+ Approach tailored to reverberant feature vector sequences
+ Long-term relations explicitely captured by reverberation model
+ Reverberation exploited for discrimination
+ Very promising results in logmelspec domain
– Inner optimization increases decoding complexity
– Implementation requires changes in decoding routines

REMOS [Sehr 2010c]

Discussion

+ Approach tailored to reverberant feature vector sequences
+ Long-term relations explicitely captured by reverberation model
+ Reverberation exploited for discrimination
+ Very promising results in logmelspec domain
– Inner optimization increases decoding complexity
– Implementation requires changes in decoding routines

Promising direction for future research

Dereverberation for Signal Enhancement

State-of-the-art

- Close to 12 dB DRR gain with $T_{60} \approx 0.7s$ (offline) with
  - 4 mics, d=1.65m, no noise (TRINICON, 2 sources)
  - 8 mics, d=2m, SNR=10 dB (MCLP)

Dereverberation for Signal Enhancement

State-of-the-art

- Close to 12 dB DRR gain with $T_{60} \approx 0.7s$ (offline) with
  - 4 mics, d=1.65m, no noise (TRINICON, 2 sources)
  - 8 mics, d=2m, SNR=10 dB (MCLP)

Challenges

- Larger distances, more reverberant rooms
- Robustness to speech-like interferers, nonstationary/diffuse noise, transient echo cancellation residuals
- Robust tracking of time-varying acoustics
- Low-latency ($\ll 1s$) and efficient real-time implementations
- Joint optimization with spectral subtraction techniques

Dereverberation as preprocessing for ASR

Dereverberation as preprocessing for ASR

Example: 20 k WSJ convolved with RIRs ($T_{60} = 0.78s$, $d = 2m$), NTT ASR

WER[%]	Preproc.	Acoustic model
85.5	none	clean speech
43.4	none	multi-condition training
26.1	1-ch derev	multi-condition training
14.2	2-ch derev	clean w/ unsuperv. speaker adaptation by MLLR

Dereverberation as preprocessing for ASR

Example: 20 k WSJ convolved with RIRs ($T_{60} = 0.78s$, $d = 2m$), NTT ASR

WER[%]	Preproc.	Acoustic model
85.5	none	clean speech
43.4	none	multi-condition training
26.1	1-ch derev	multi-condition training
14.2	2-ch derev	clean w/ unsuperv. speaker adaptation by MLLR

Challenges for approaching close-talk performance

- Transition from reverberated signals to real recordings
- Self-adaptation to changing acoustics and front-ends, including
  - variable number and changing, unconstrained positions of talkers
  - different nodes in distributed microphone arrays
- Joint optimization with ASR methods to handle reverberation and noise

Reverberation-specific ASR Techniques

State-of-the-art

Reverberation-specific ASR Techniques

State-of-the-art

- Feature-based techniques
  - account for the inter-frame relations caused by dispersion
  - efficiently exploit predictability of reverberation

Reverberation-specific ASR Techniques

State-of-the-art

- Feature-based techniques
  - account for the inter-frame relations caused by dispersion
  - efficiently exploit predictability of reverberation
- Model-based techniques
  - could not yet show their full potential, as
  - framewise adaptation and optimization is computationally complex

Reverberation-specific ASR Techniques

State-of-the-art

- Feature-based techniques
  - account for the inter-frame relations caused by dispersion
  - efficiently exploit predictability of reverberation
- Model-based techniques
  - could not yet show their full potential, as
  - framewise adaptation and optimization is computationally complex
- Decoder-based techniques
  - compromise between the above regarding complexity

Reverberation-specific ASR Techniques

State-of-the-art

- Feature-based techniques
  - account for the inter-frame relations caused by dispersion
  - efficiently exploit predictability of reverberation
- Model-based techniques
  - could not yet show their full potential, as
  - framewise adaptation and optimization is computationally complex
- Decoder-based techniques
  - compromise between the above regarding complexity

Outlook

- Integration into state-of-the art ASR systems
  - expected soon for signal enhancement- and feature-based methods
  - model-based methods must become more efficient for widespread use

Concluding remarks

Dereverberation, the ’Holy Grail’ of Acoustic Signal Processing?

Concluding remarks

Dereverberation, the ’Holy Grail’ of Acoustic Signal Processing?

- Blind deconvolution of the acoustic paths seems to come closer

Concluding remarks

Dereverberation, the 'Holy Grail' of Acoustic Signal Processing?

- Blind deconvolution of the acoustic paths seems to come closer
- Less ambitious algorithms are also effective and their progress follows the typical DSP objectives
  - increase algorithmic performance and robustness
  - reduce computational load
  - integrate with other functionalities

Concluding remarks

Dereverberation, the 'Holy Grail' of Acoustic Signal Processing?

- Blind deconvolution of the acoustic paths seems to come closer
- Less ambitious algorithms are also effective and their progress follows the typical DSP objectives
  - increase algorithmic performance and robustness
  - reduce computational load
  - integrate with other functionalities

As a follow-up to the CHIME Challenge 2011

$\Rightarrow$ **Next Challenge for Reverberation-robust Speech Processing is underway!**

Acknowledgements

We are especially grateful to

- Dr. Keisuke Kinoshita, Dr. Marc Delcroix, Dr. Shoko Araki, Dr. Mehrez Souden, and Dr. Takaaki Hori (NTT)
- Dr. Herbert Buchner, Edwin Mabande and Lutz Marquardt (formerly LMS)
- Roland Maas and Christian Hofmann (LMS)

for their contributions to the course material

Acknowledgements

We are especially grateful to

- Dr. Keisuke Kinoshita, Dr. Marc Delcroix, Dr. Shoko Araki, Dr. Mehrez Souden, and Dr. Takaaki Hori (NTT)
- Dr. Herbert Buchner, Edwin Mabande and Lutz Marquardt (formerly LMS)
- Roland Maas and Christian Hofmann (LMS)

for their contributions to the course material and wish to acknowledge the support of parts of the LMS by

- Deutsche Forschungsgemeinschaft (DFG) under contract number KE 890/4-1

ご清聴ありがとうございました