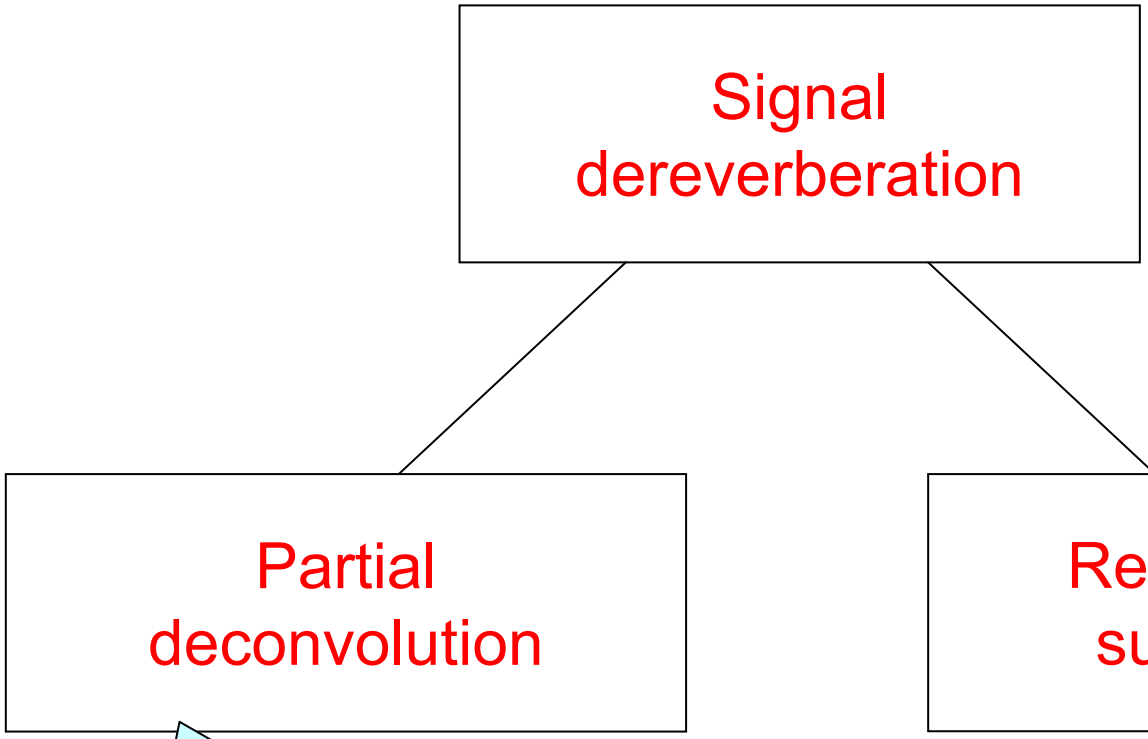
Part II.
Multichannel blind inverse filtering

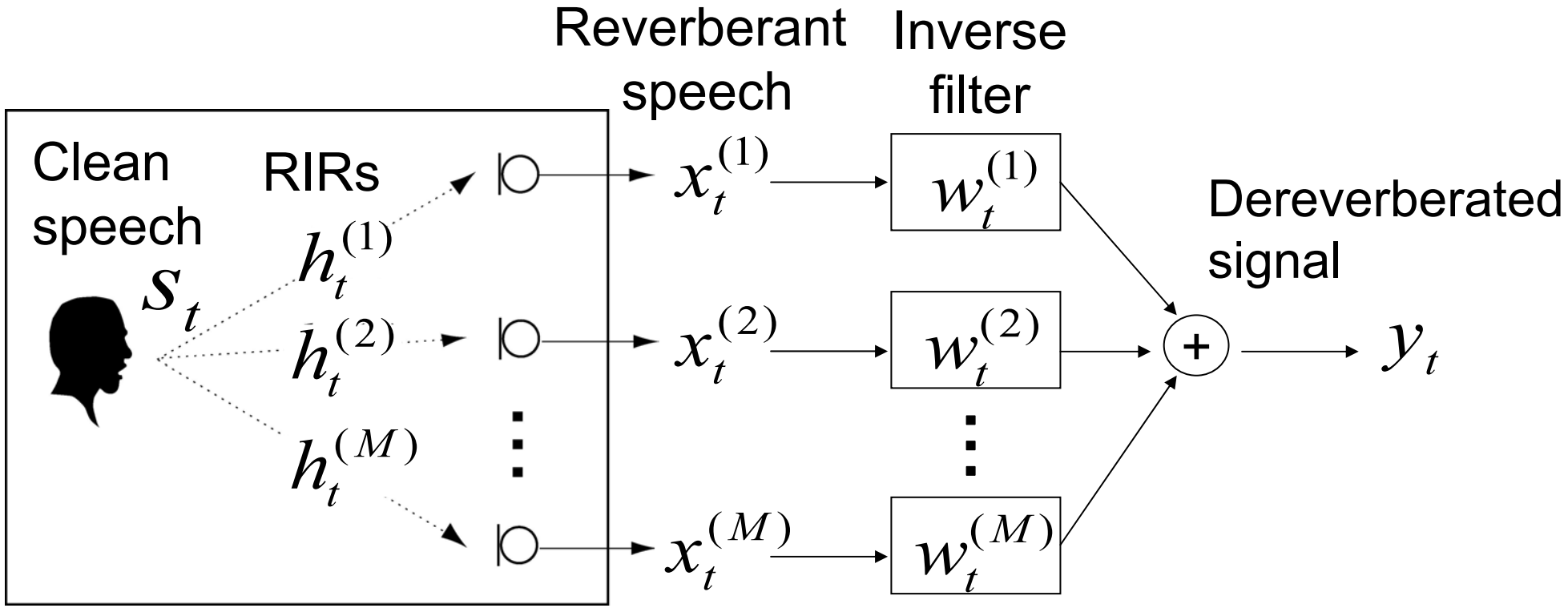
Two approaches for signal dereverberation



"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

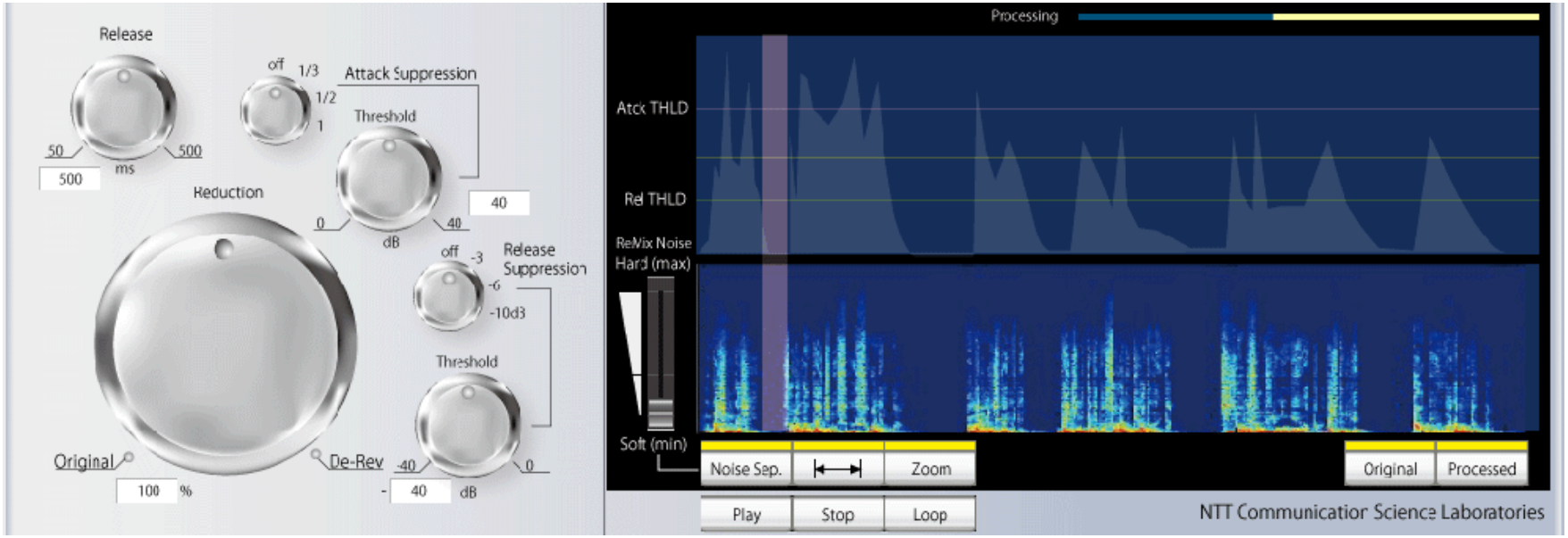
[Movies/TV creation]

Step1:
Sound&video recording (on location)

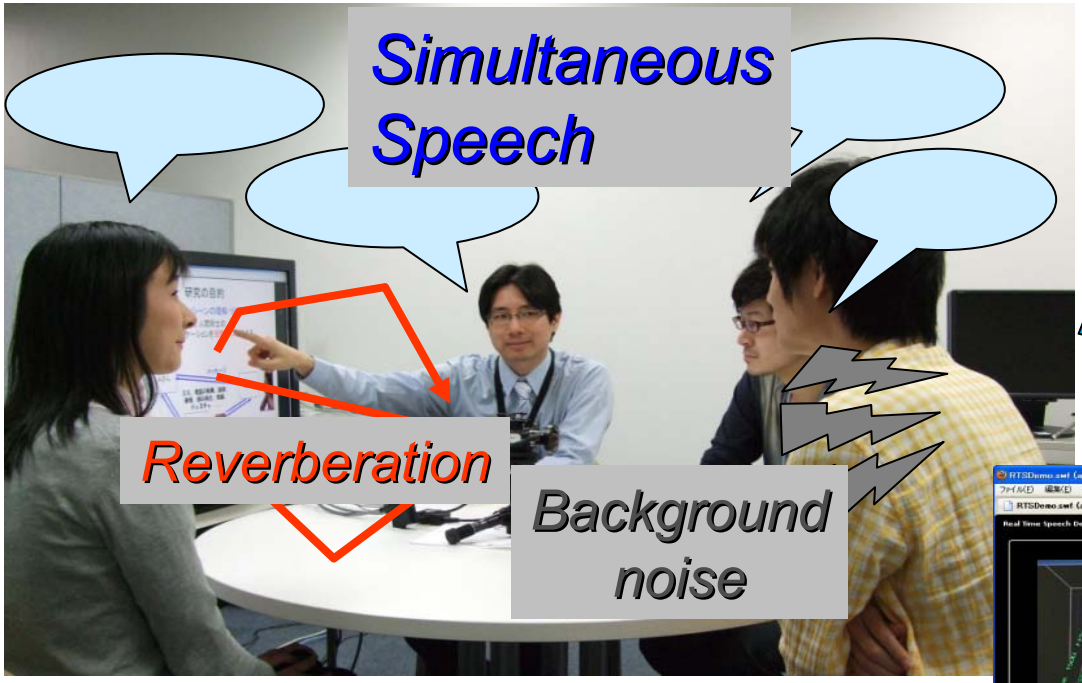
Step2:
Audio post-production
(de-noising, ***de-reverb***, sound effects)

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]



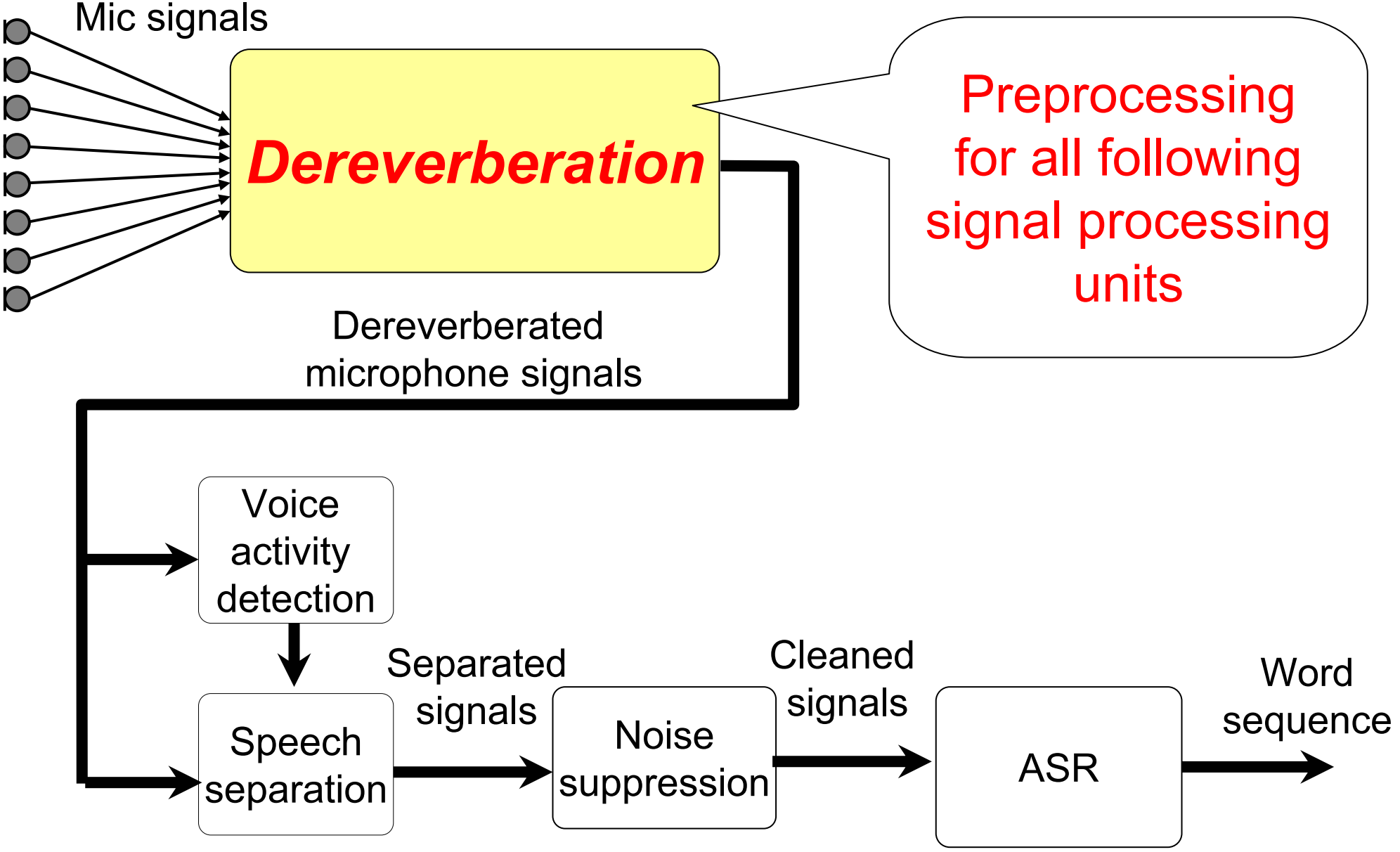
Recognize speech and other audio events

Real-time Meeting Browser

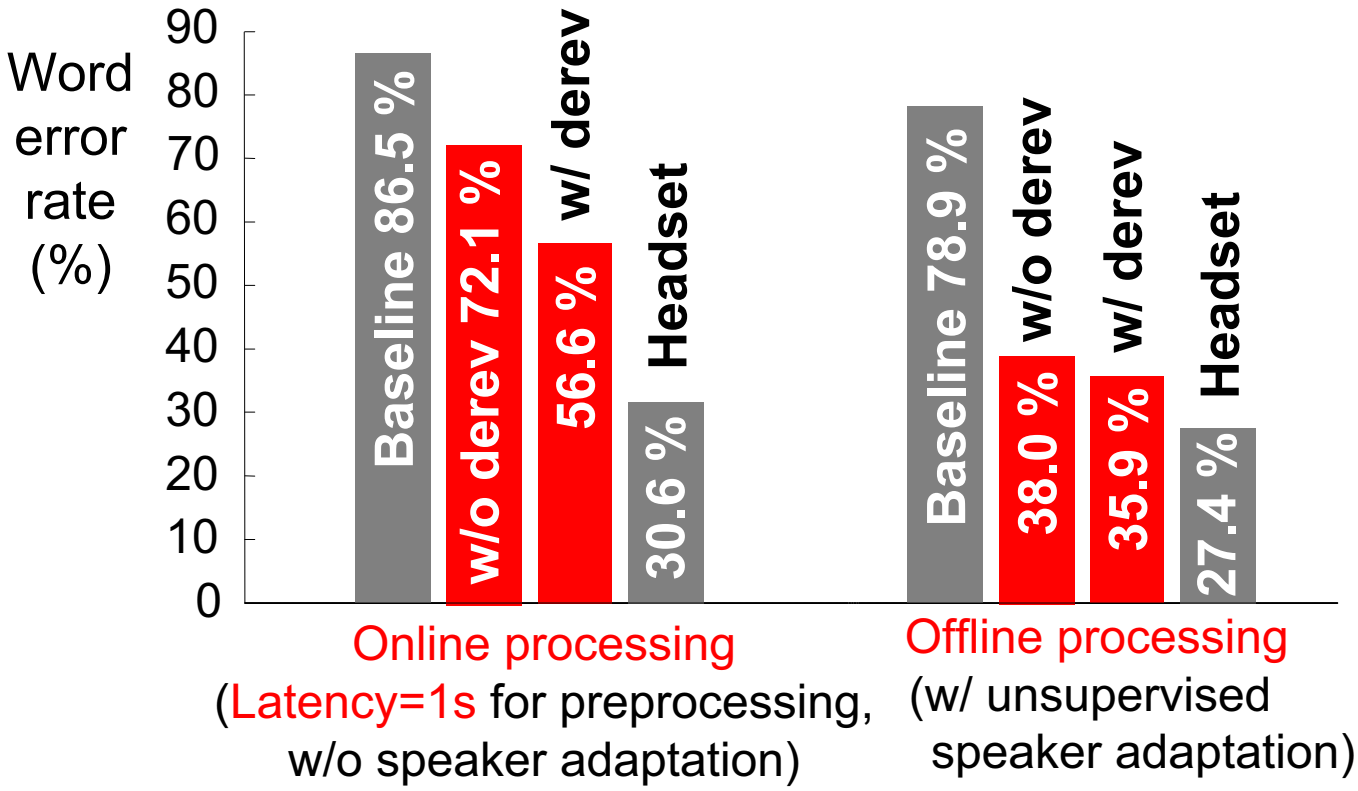
Who Spoke When, What, To-whom and How?

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

Online/offline processing flow of meeting recognition



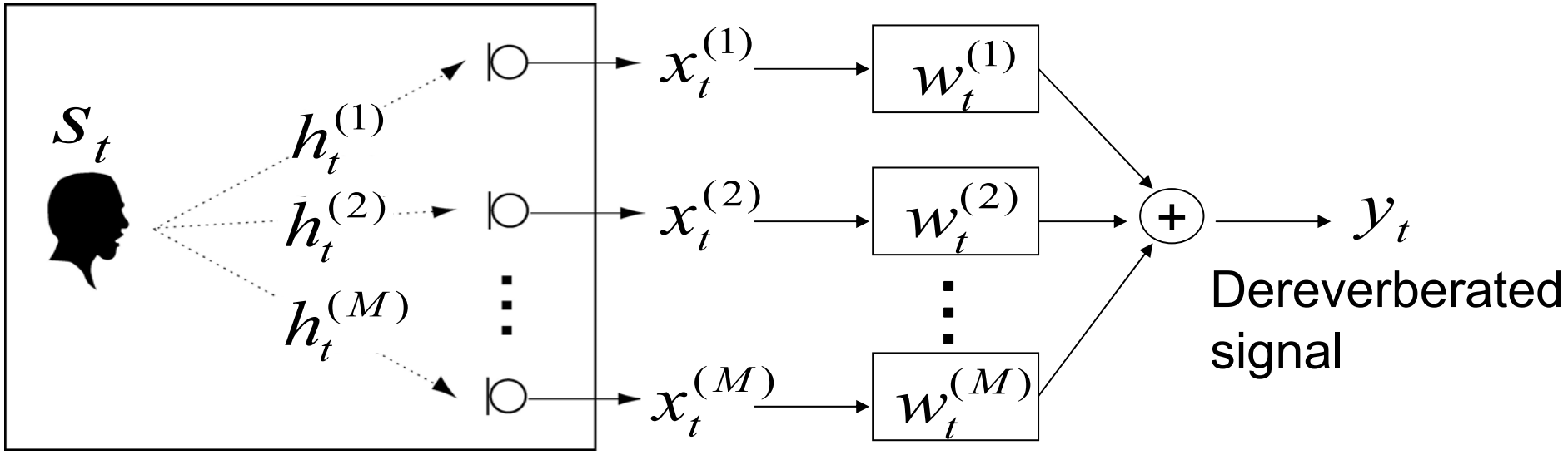
ASR performance w/ and w/o dereverberation



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

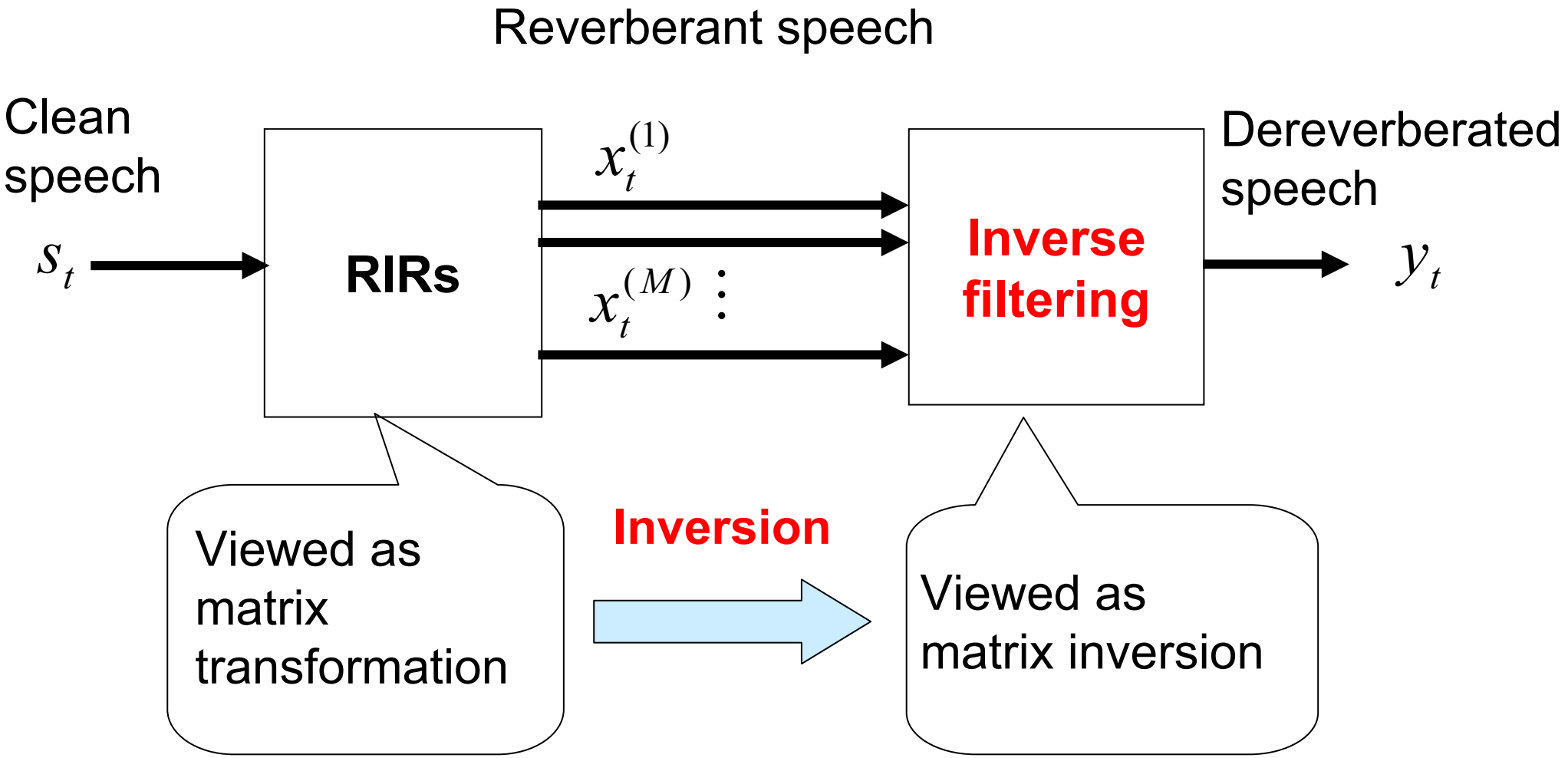
Assume ***non-blind processing*** for analysis purpose

- **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

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

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

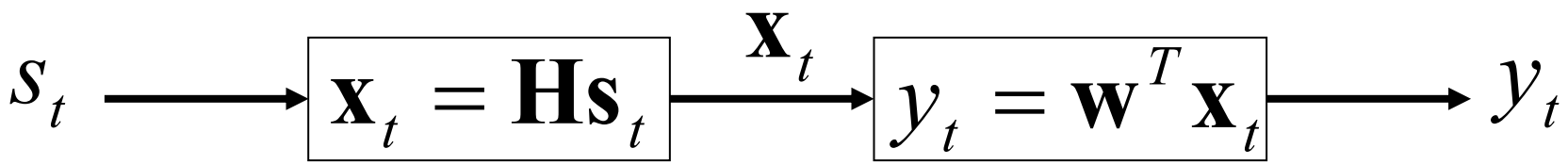
Multichannel filtering

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

$$\mathbf{x}_t$$

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

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

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

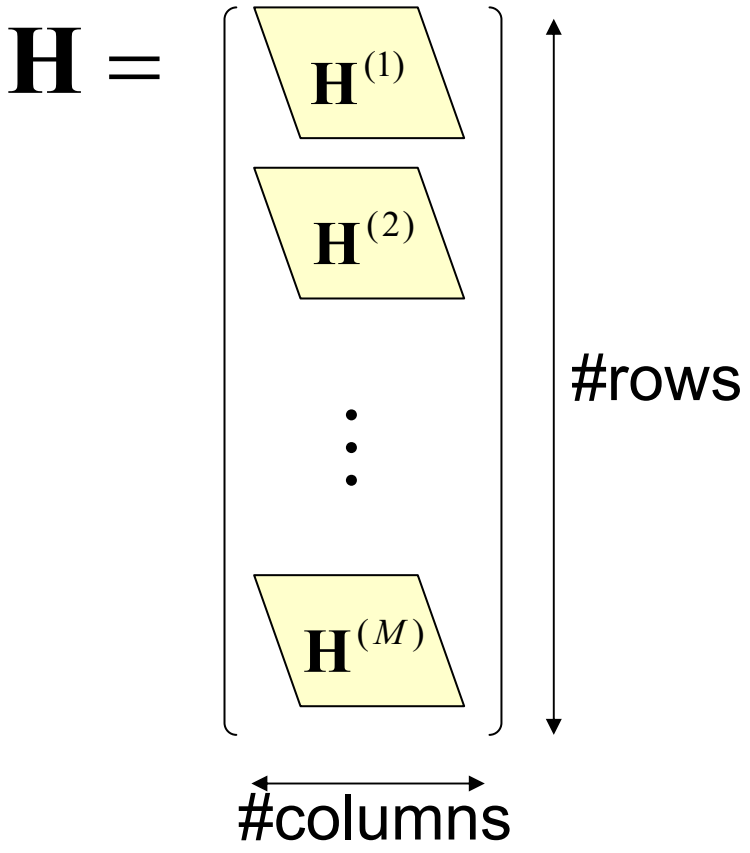
M (#mics) > 1 is required for single source case

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

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

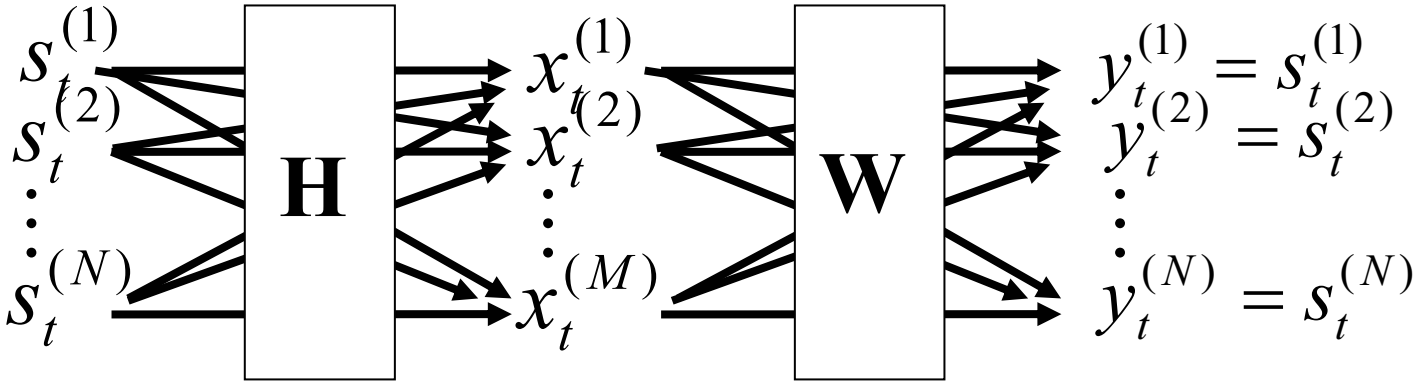
  and all columns are linearly independent
- In the case of single source, (#rows of $\mathbf{H}$) >= (#columns of $\mathbf{H}$) is satisfied if and only if

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

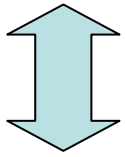


Generalization to N sources × M microphones case

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

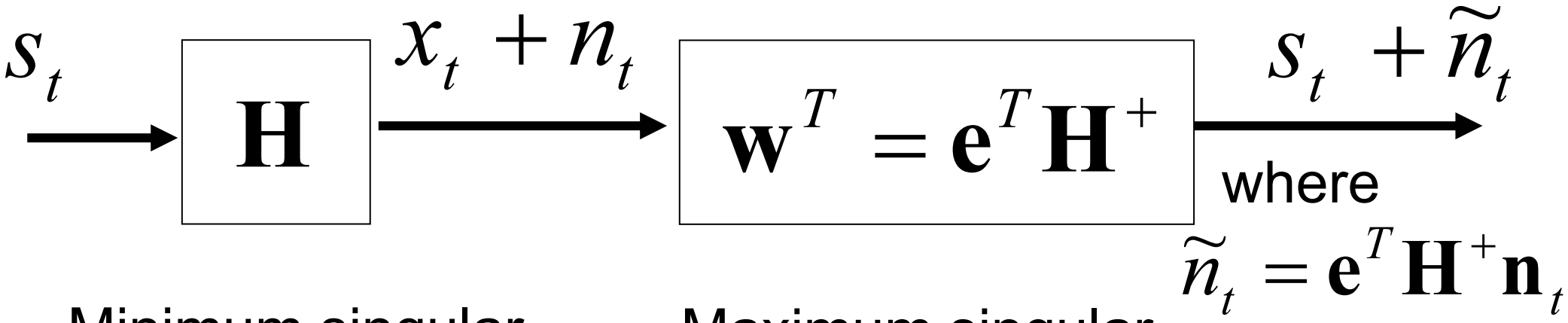
Noise-free reverberant case

- 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

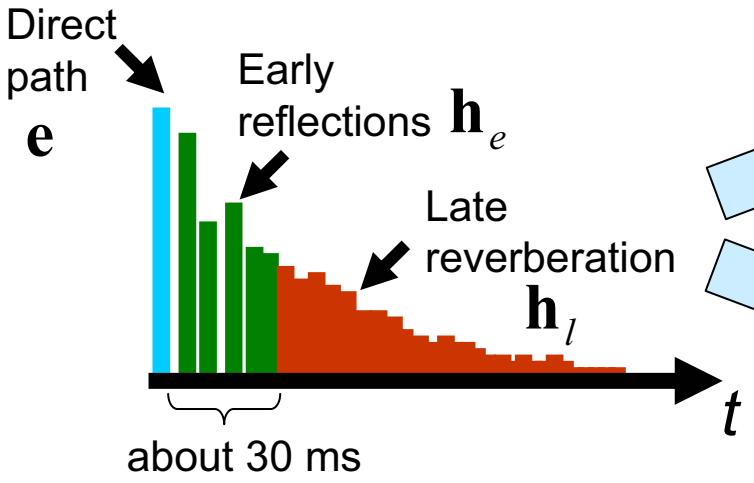
  - It can reduce the maximum singular value $\lambda_{\max}^{\text{inv}}$ of $\tilde{\mathbf{H}}^+$

⇒ 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

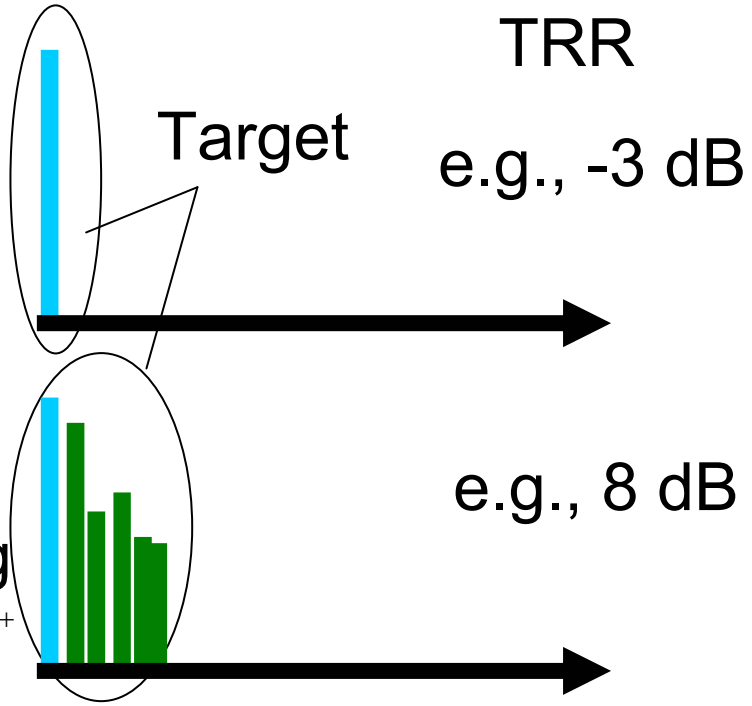


Inverse filtering

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

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

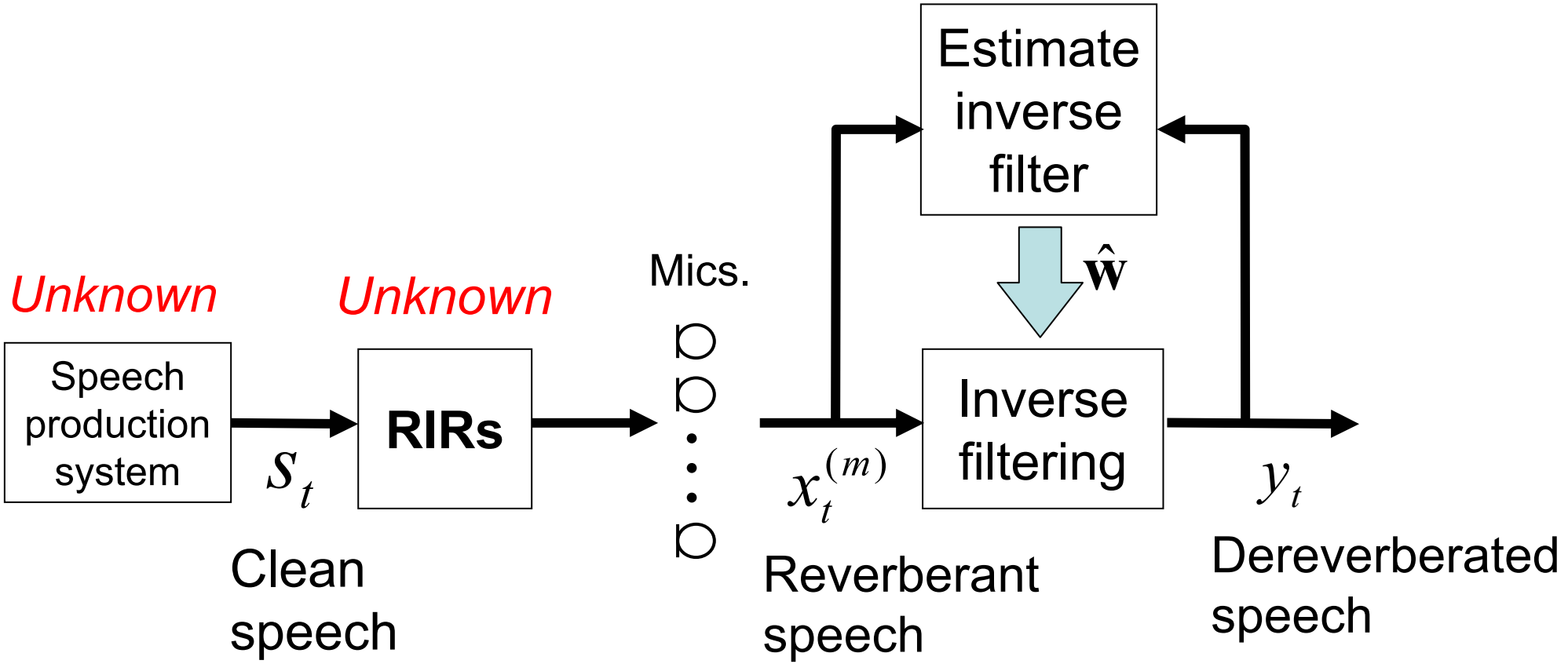
- Example applications
  - Professional audio post production
  - Meeting recognition with microphone arrays

- 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

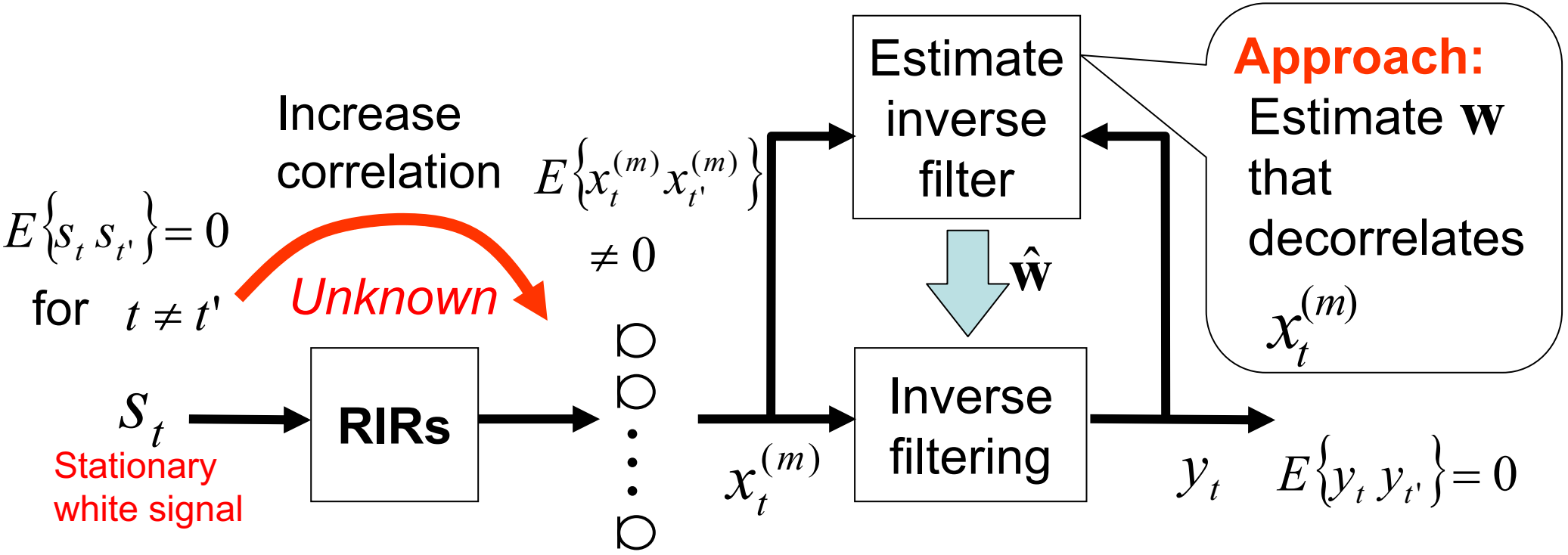
Blind inverse filtering based dereverberation



Two approaches

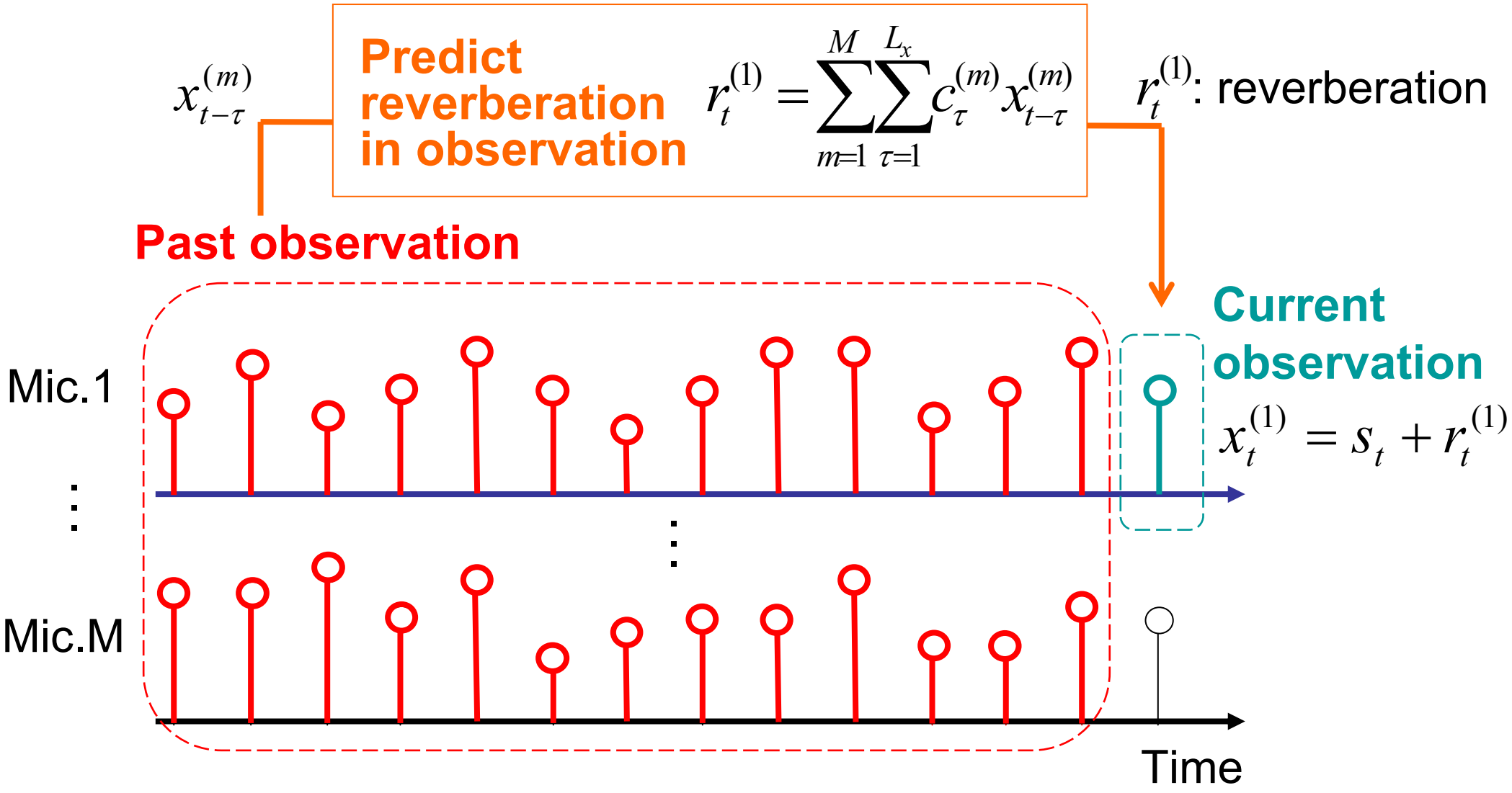
- RIR estimation + RIR inversion
- ***Direct estimation of inverse filter***

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

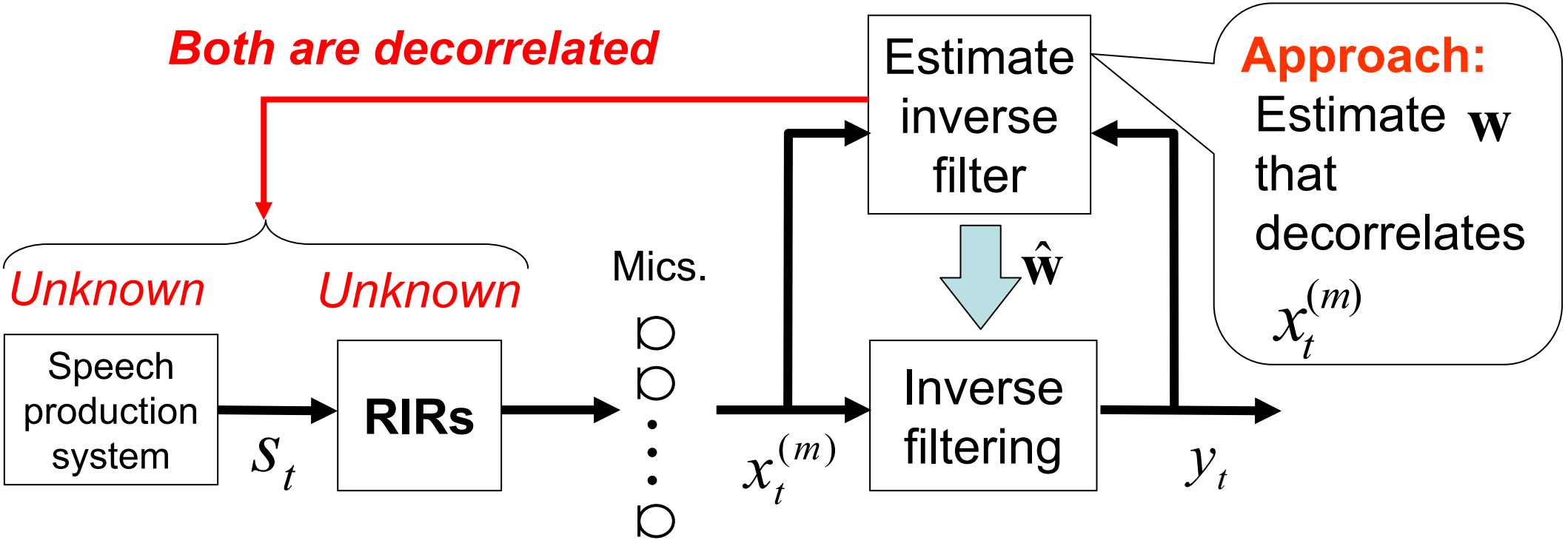
$$\sum_{t=1}\left(z_t^{(1)} - \mathbf{z}_{t-1}^T\mathbf{c}\right)^2 = \sum_{t=1}\left(x_t^{(1)} - \mathbf{x}_{t-1}^T\mathbf{c}\right)^2 + \sum_{t=1}\left(n_t^{(1)} - \mathbf{n}_{t-1}^T\mathbf{c}\right)^2$$

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

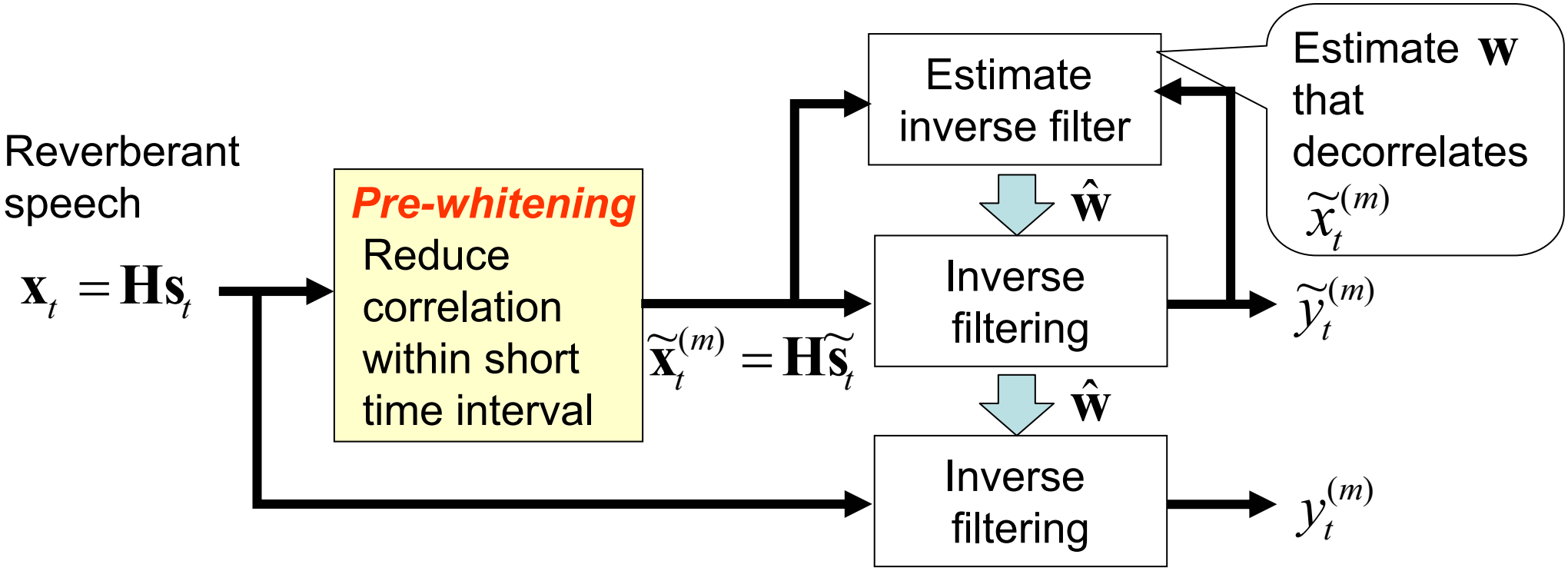
- ***Pre-whitening + decorrelation***
  - Second-order statistics (SOS): [Gaubitch 2003], [Furuya 2007], [Triki 2007]
  - Higher-order statistics (HOS): [Gillespie 2001]
- ***Channel shortening***
  - [Gillespie 2003], [Kinoshita 2009]

  } 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

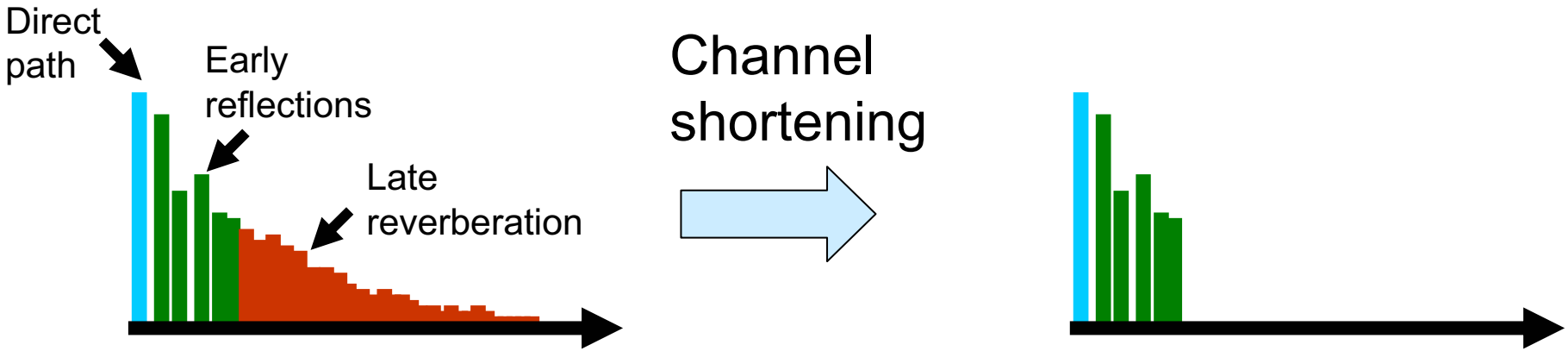


- A typical method for pre-whitening
  - Low-dimensional (e.g., 12-dim) single channel linear prediction often used

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

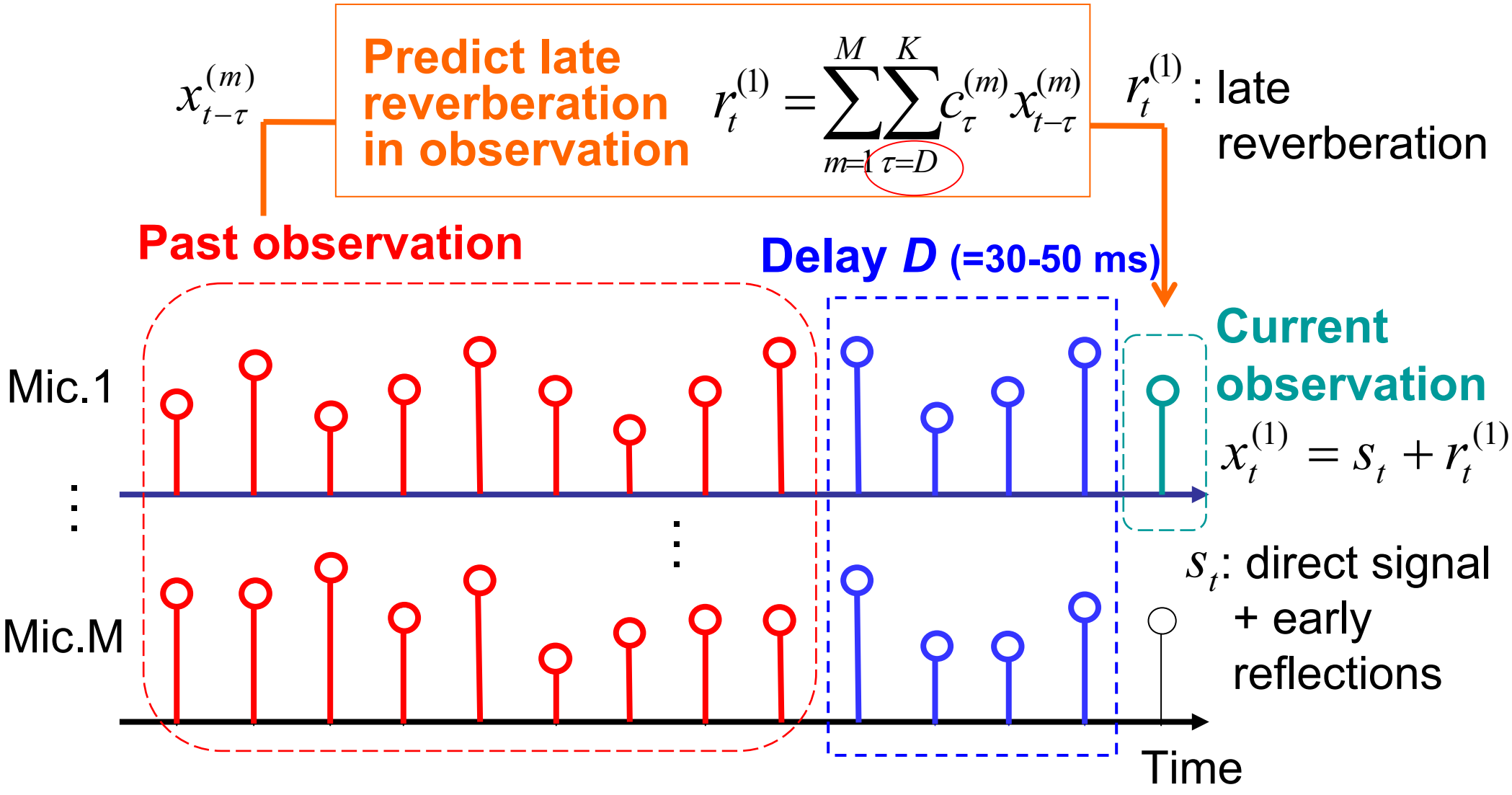
- Introduce constraints so that dereverberation reduces only late reverberation



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]

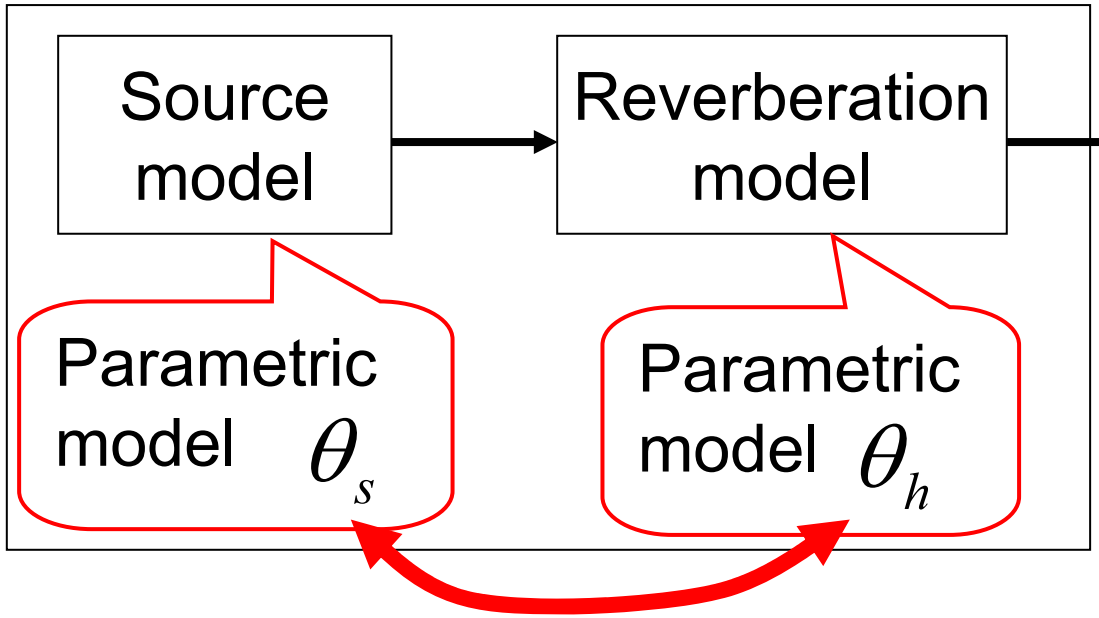
  } Duration of auto-correlation

- ***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.

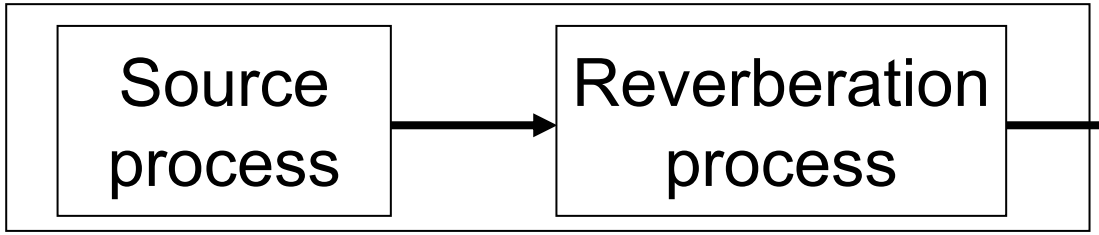
Model of generative system



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

Parameter estimation by

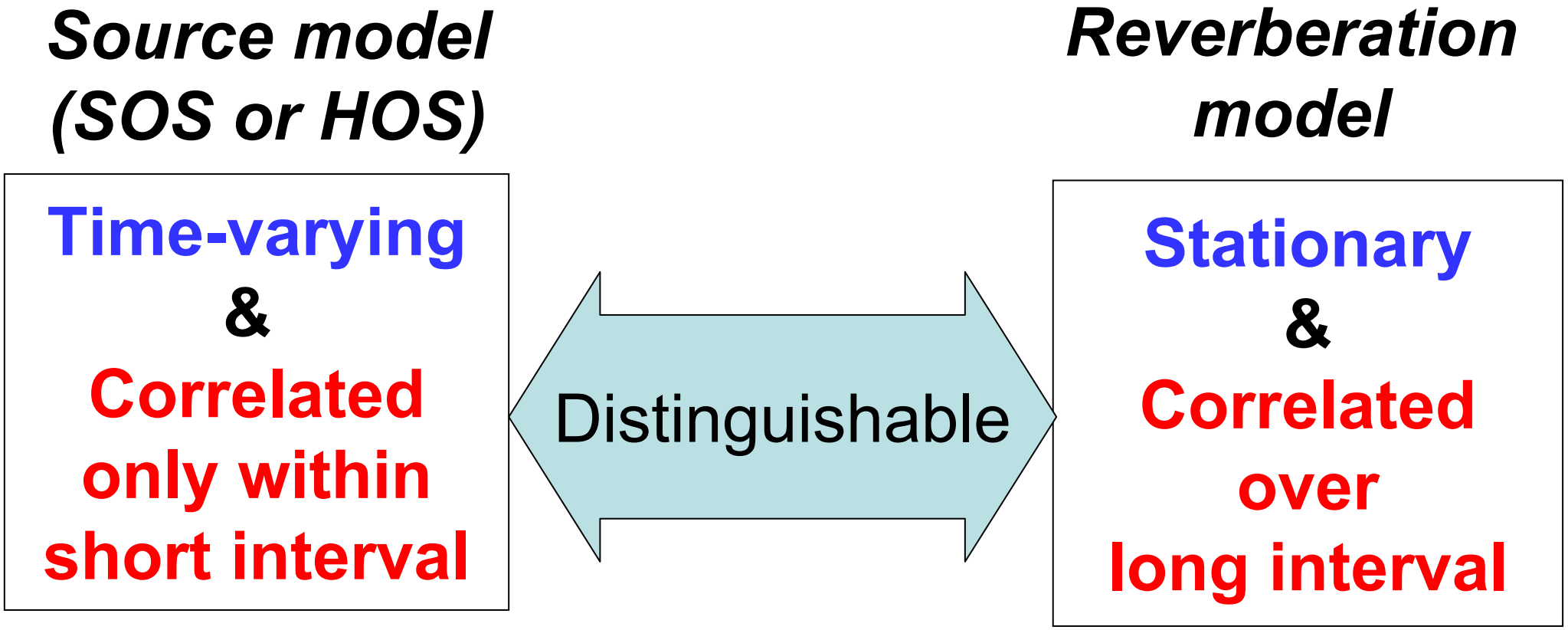
- **Likelihood maximization**
  [Hopgood 2003], [Yoshioka 2007], [Nakatani 2008]
- **Kullback-Leibler divergence minimization**
  [Buchner (TRINICON) 2010]



x_t Reverberant observation

Unknown true generative system

Models for source process and reverberation process



Multichannel blind partial deconvolution (MCBPD) by TRINICON

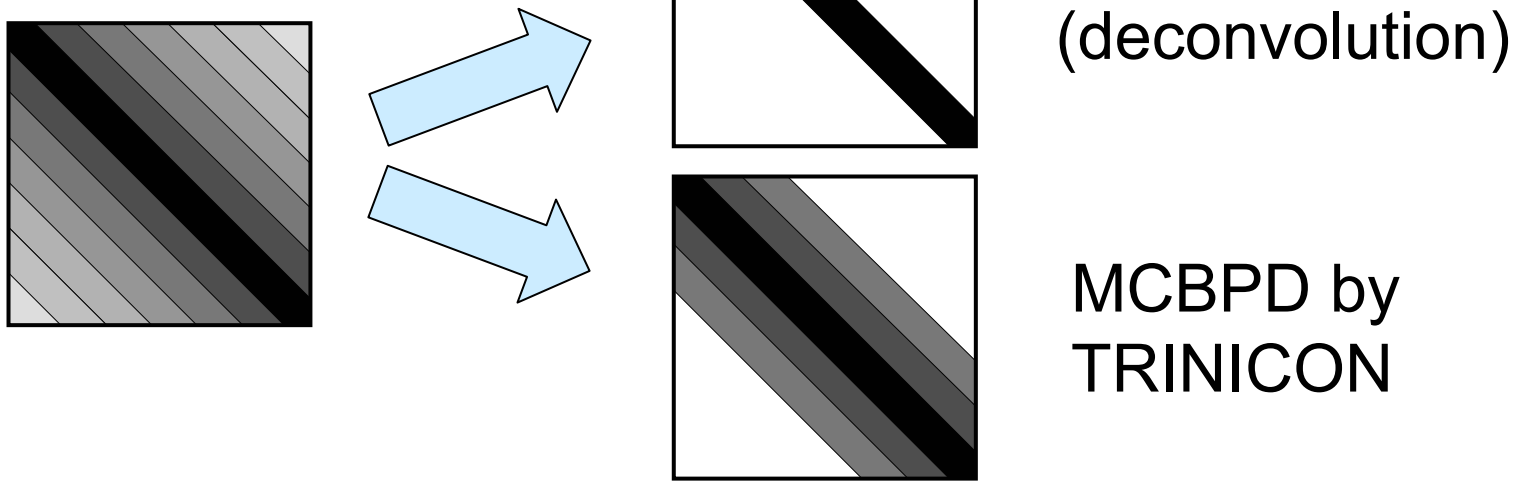
Cost function for SOS-TRINICON [Buchner 2010]

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

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

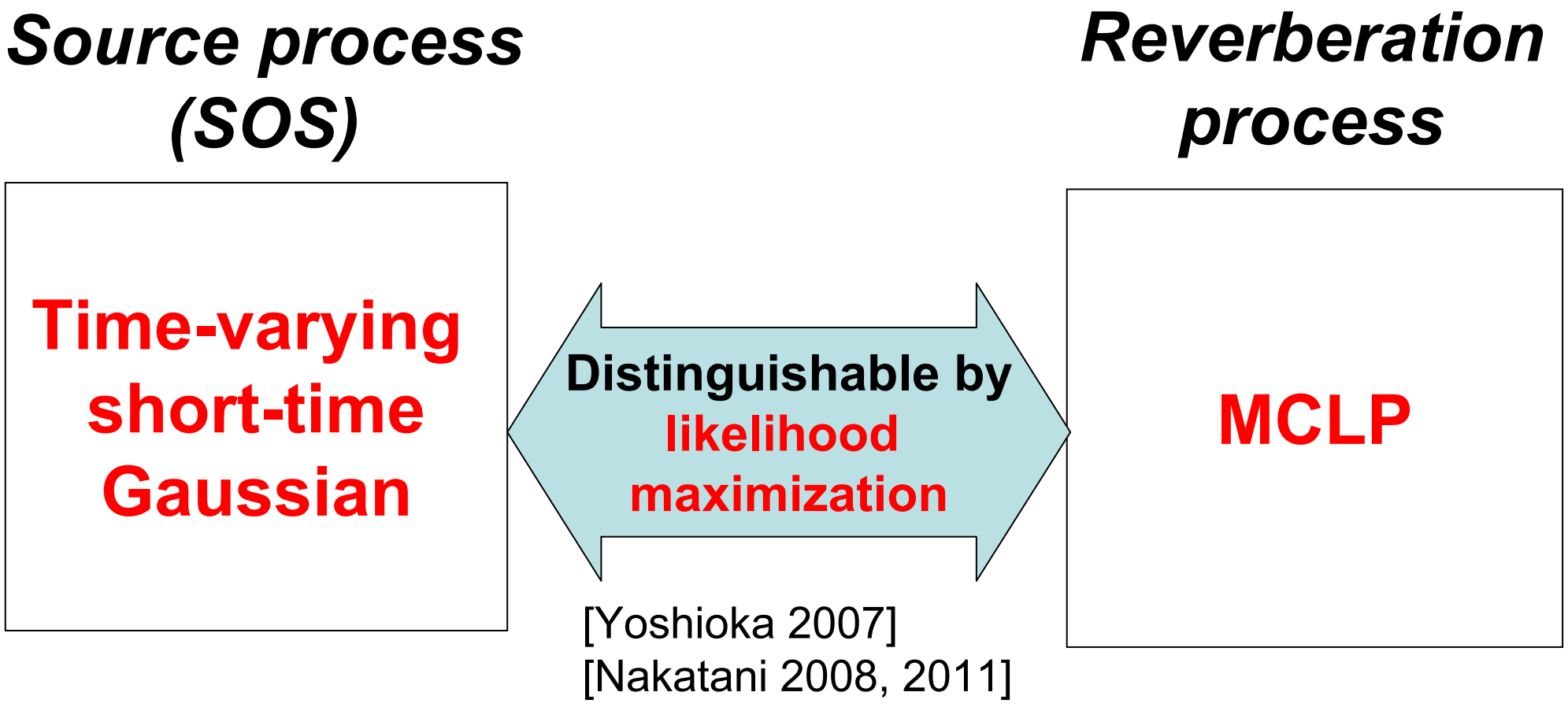
- Example applications
  - Professional audio post production
  - Meeting recognition with microphone arrays

- 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 $\{\bar{\mathbf{x}}_t\}$

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

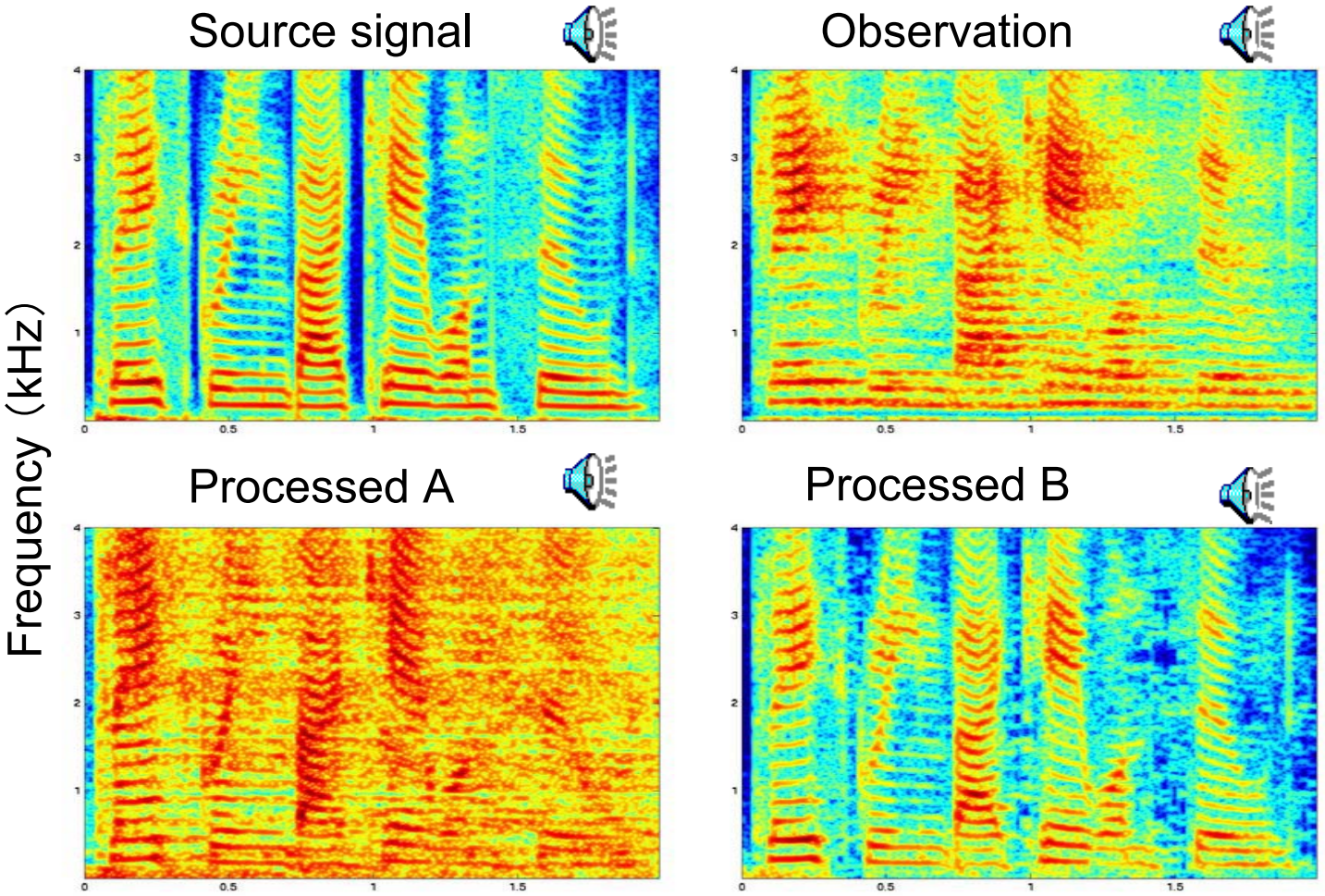
2. Calculate autocorrelation matrix of $\hat{\bar{\mathbf{s}}}_t$

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

$\{\hat{\bar{\mathbf{s}}}_t\}$ ↓

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

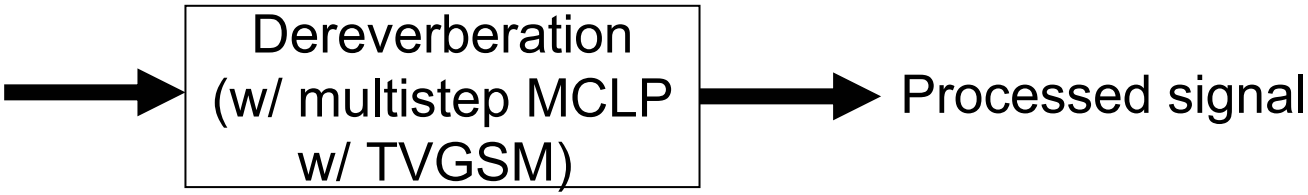
A few seconds of observation are sufficient for dereverberation

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

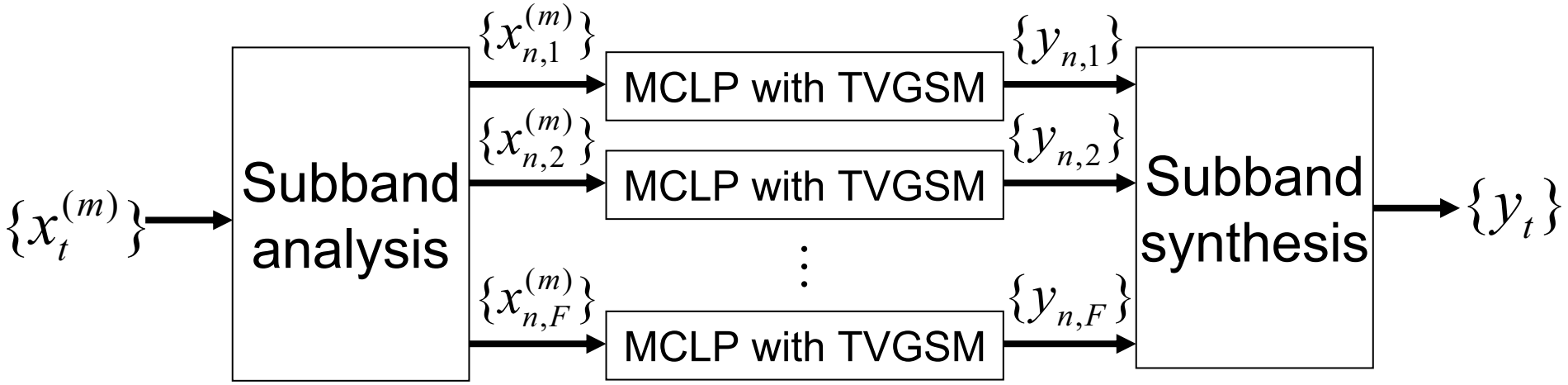
Noise: additive white noise (reproduced and recorded by 8 mics)

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

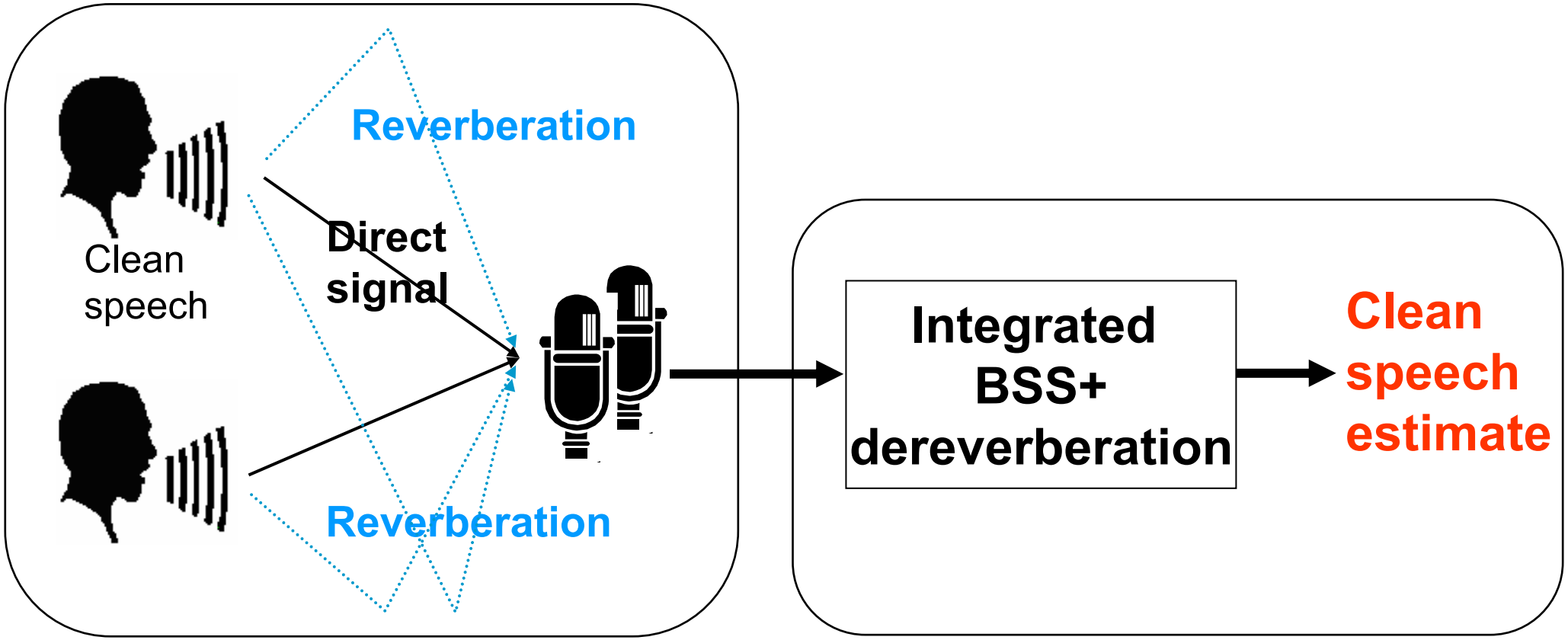
- Example applications
  - Professional audio post production
  - Meeting recognition with microphone arrays

- 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**

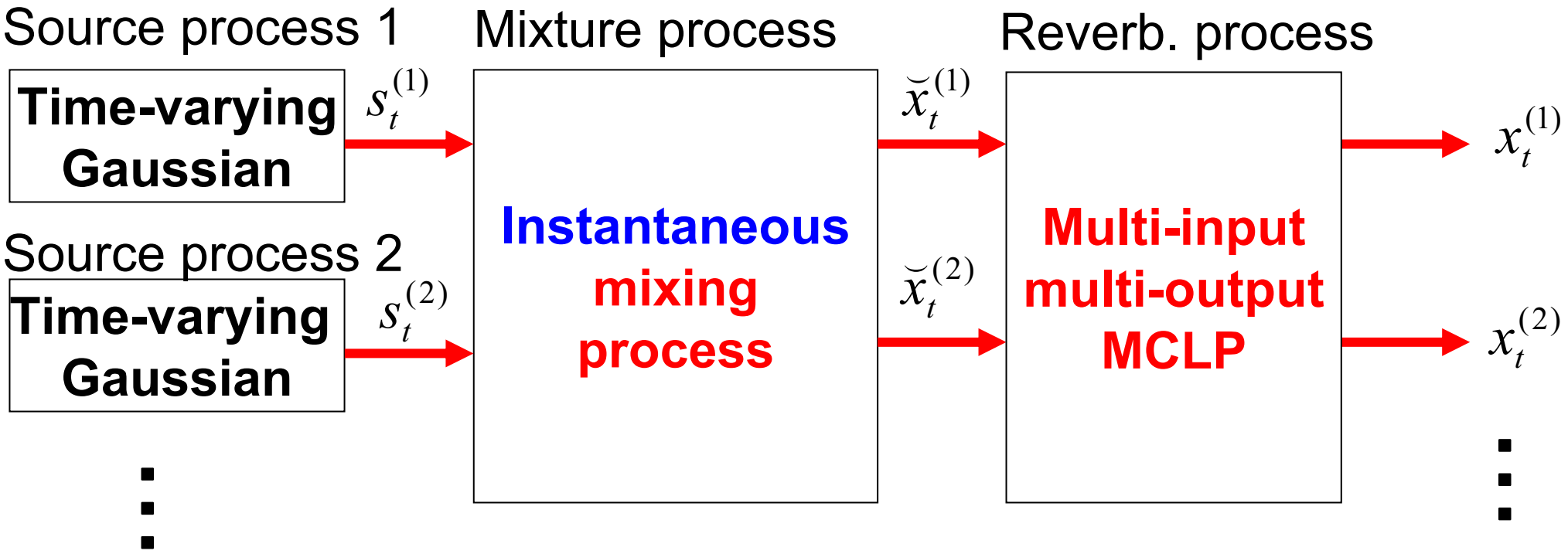
BSS+dereverberation



Approaches:

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

Generative model for reverberant sound mixture

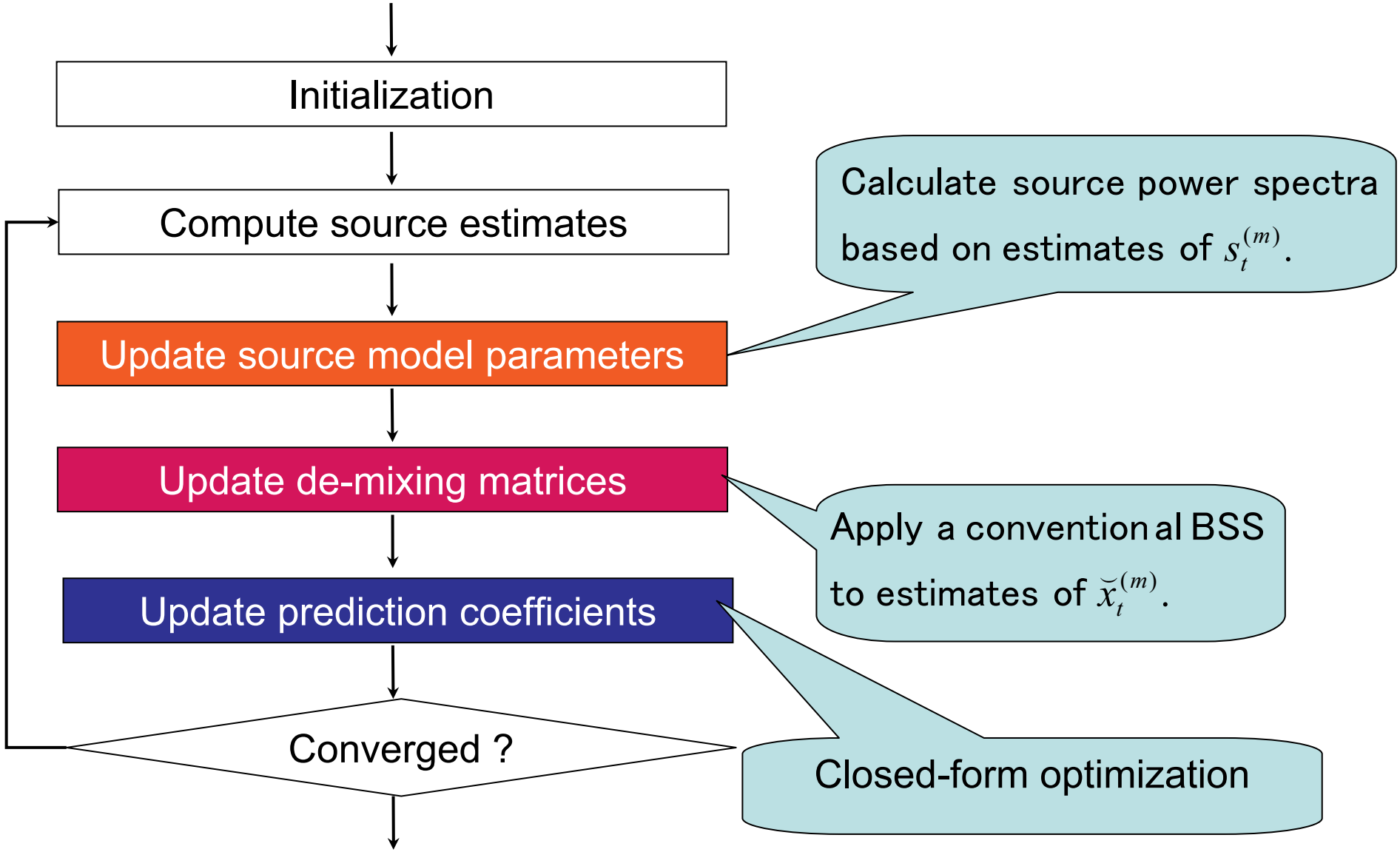


$\breve{x}_t^{(m)}$: non-reverberant mixture

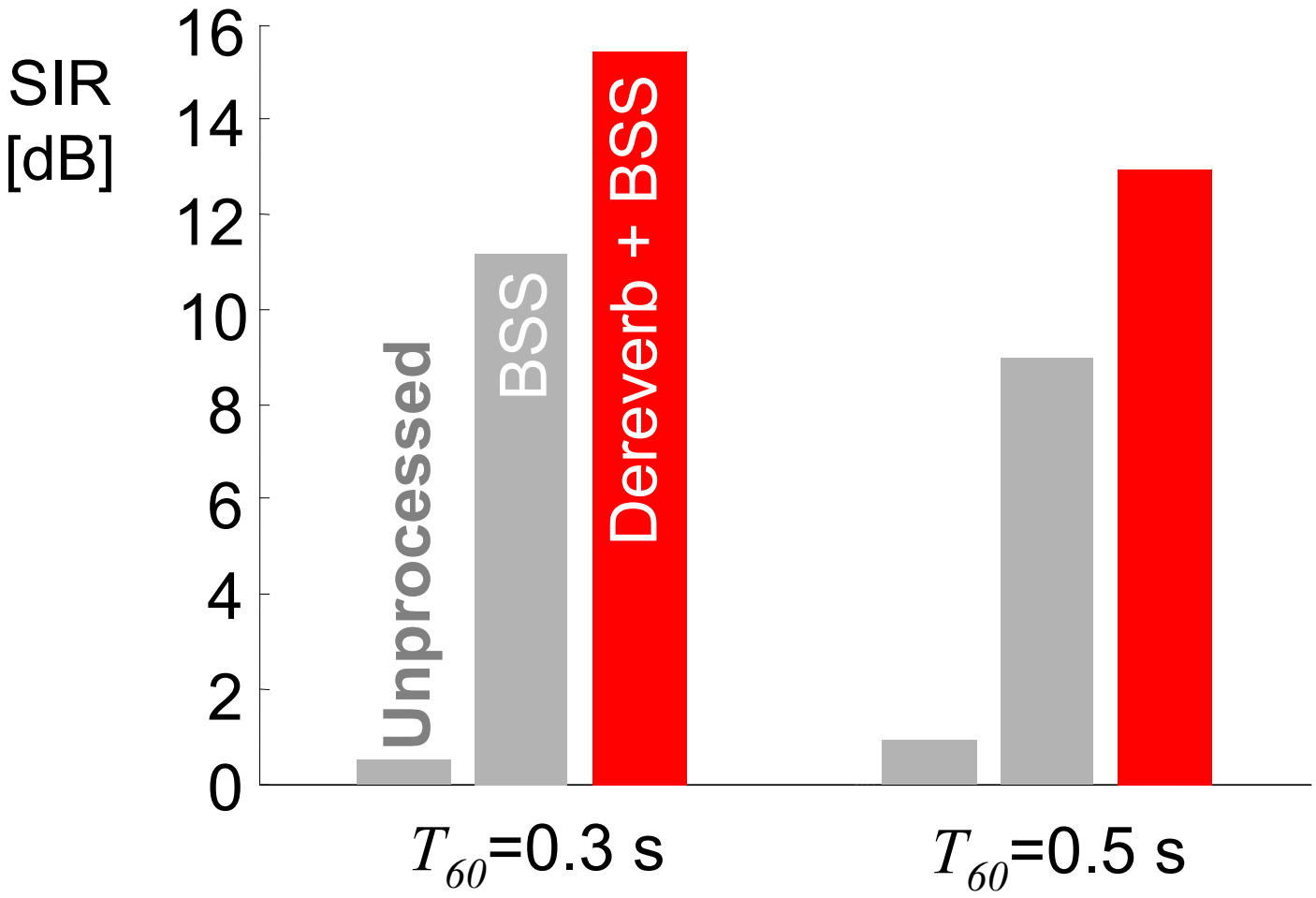
$x_t^{(m)}$: reverberant mixture

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

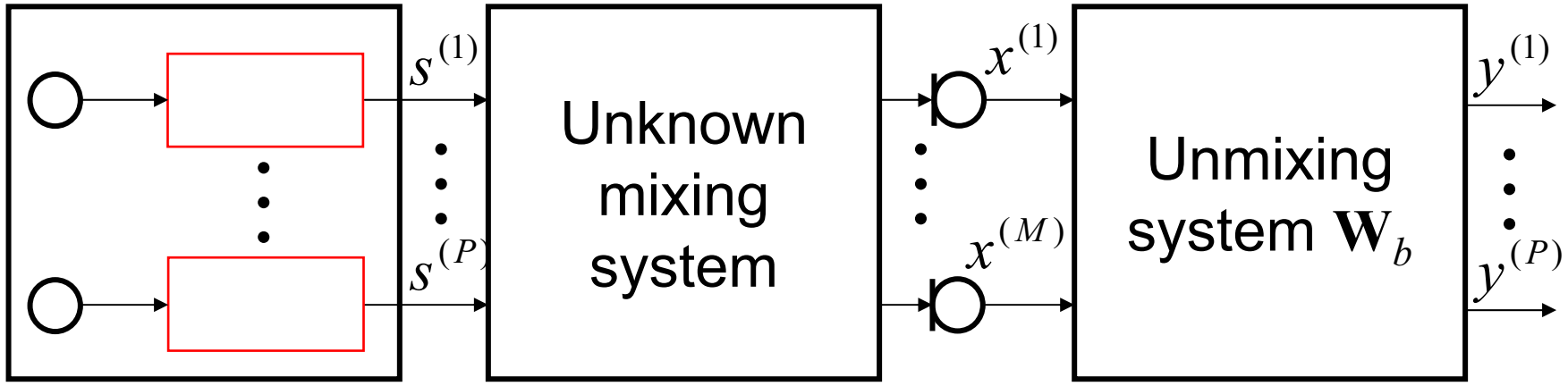
Recording: 1 to 8 s (average: 3.5 s)

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]

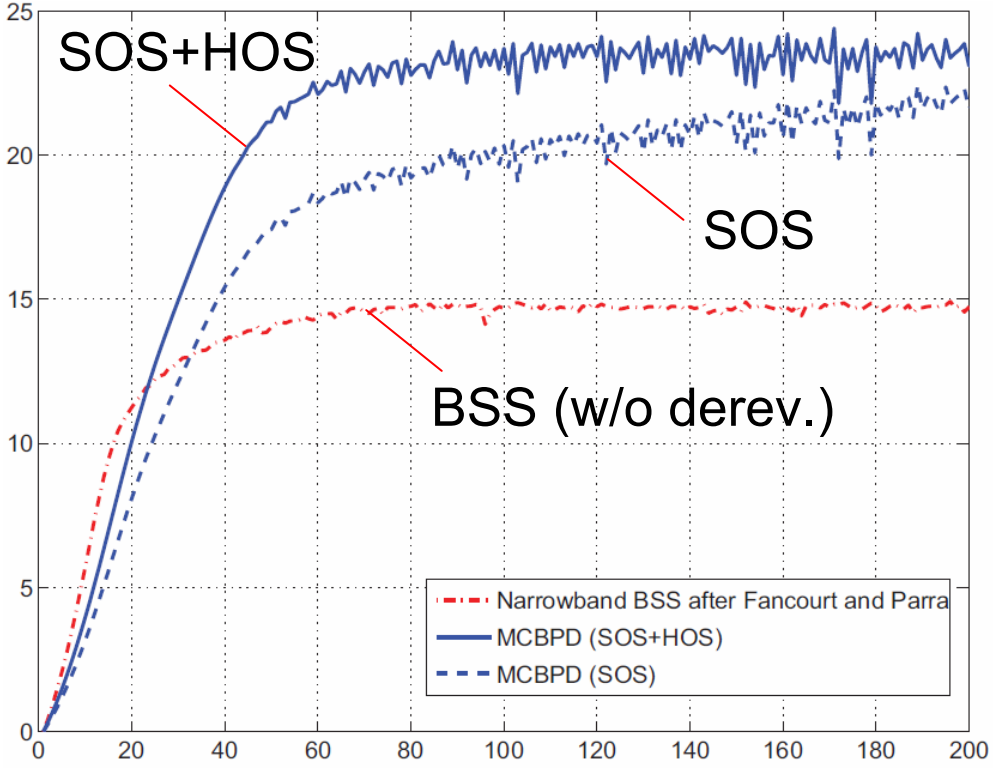
$$\mathcal{J}(\mathbf{W}_b) = -\sum_{i=0}^{\infty} \beta(i,b) \sum_{j=0}^{N_0} \left\{ \log(\hat{p}_{s,PD}(\mathbf{y}(i,j))) - \log(\hat{p}_{y,PD}(\mathbf{y}(i,j))) \right\}$$

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

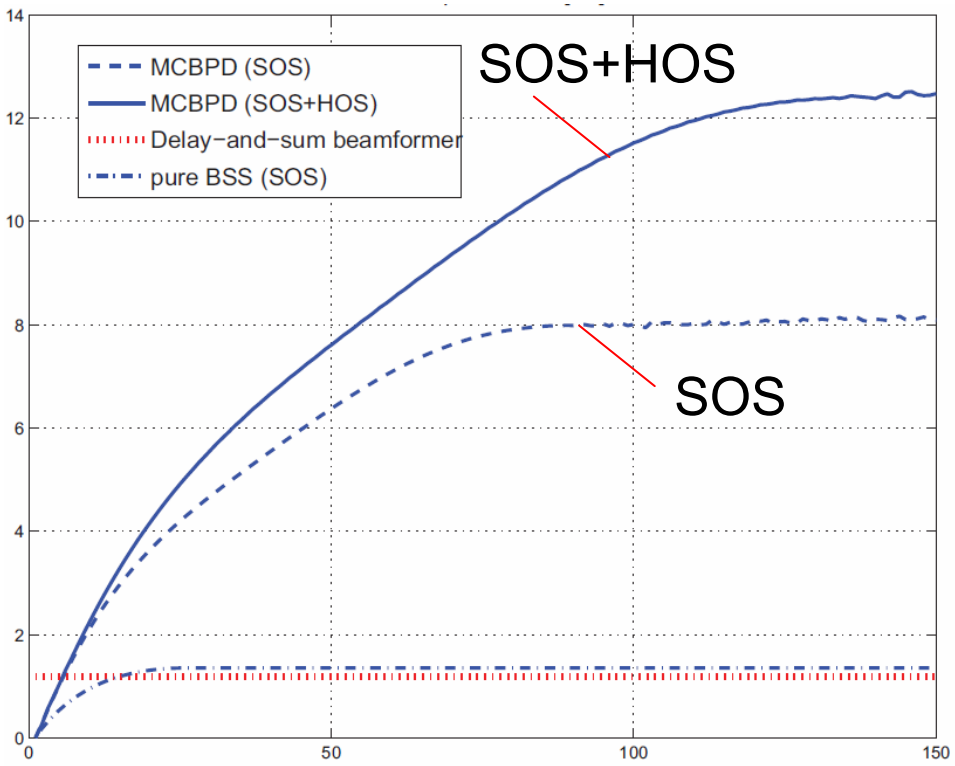
Comparison of SOS and HOS by TRINICON [Buchner 2010]

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