Microphone Array Processing

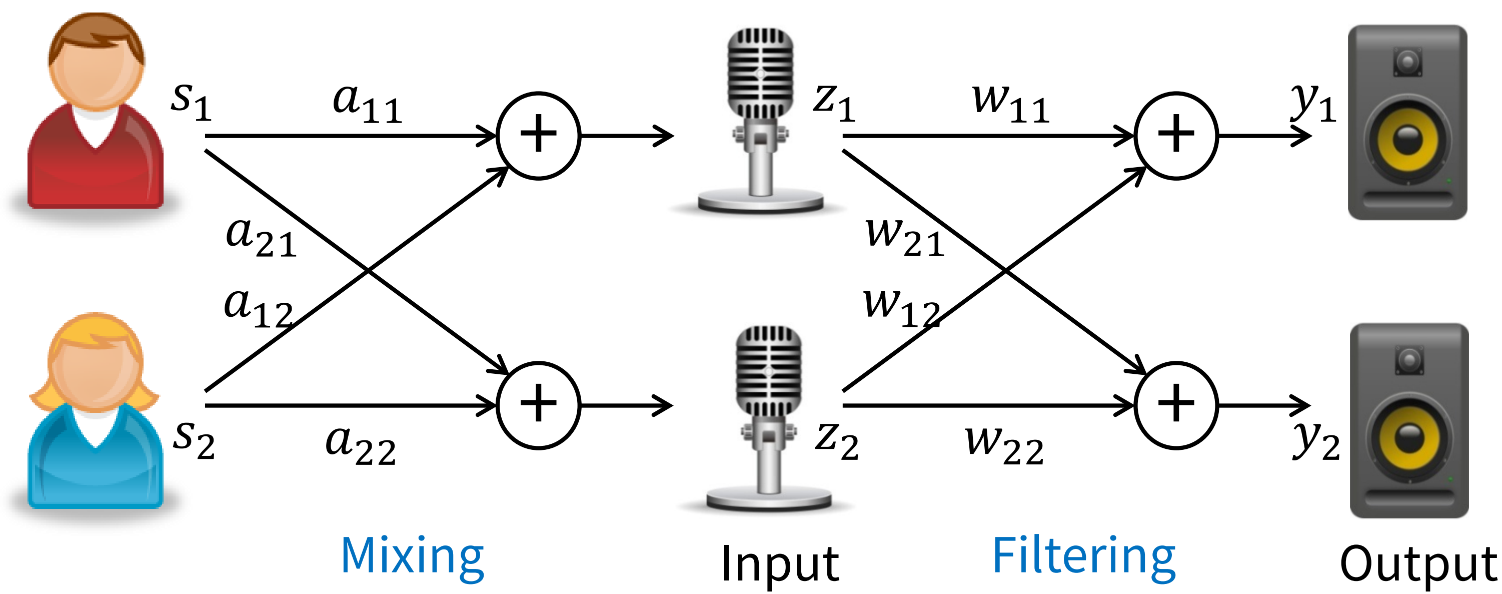
Graduate School of Informatics
Kyoto University
Kazuyoshi Yoshii
yoshii@kuis.kyoto-u.ac.jp

Microphone Array Processing

- A fundamental technique for various studies
  - Speech recognition & cocktail-party effect
    - It is important to selectively listen to utterances of interest even if we make conversation in a noisy environment
  - Robot audition
    - Robots should use their own ears for listening to sounds
    - Individual sound sources should be localized and separated
  - Analysis of recorded speech communication
    - Speaker identification
    - Voice activity detection for each speaker
    - Noise/reverberation reduction

Goal

- We aim at sound source separation and localization
  - Input: $z_1, z_2, \cdots, z_N$ Output: $y_1, y_2, \cdots, y_M$ ($\approx s_1, s_2, \cdots, s_M$)
    - Mixing process: sources $s_1, s_2, \cdots, s_M$ → observations $z_1, z_2, \cdots, z_N$
    - Two settings: $\boldsymbol{A}$ is given (non-blind) ↔ $\boldsymbol{A}$ is not given (blind)

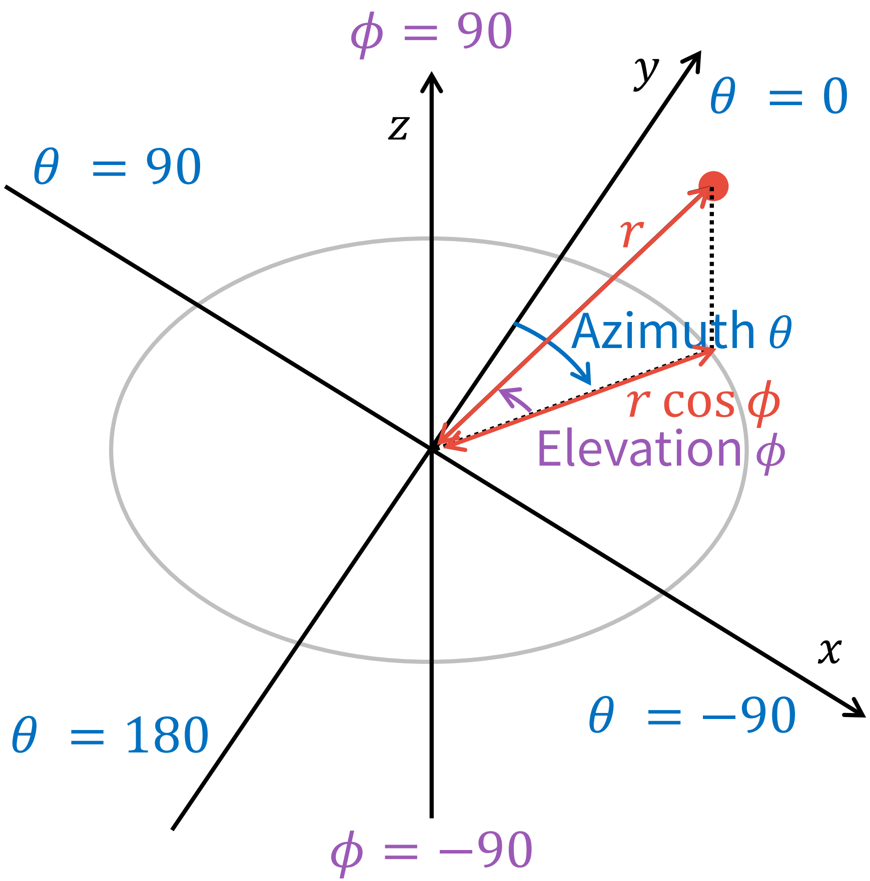


Overview

- Two major approaches to microphone array processing
  - Non-blind setting
    - Beamformer
    - MUSIC (multiple signal classification)
  - Blind setting
    - Independent component/vector analysis (ICA/IVA)
    - Multi-channel nonnegative matrix factorization (NMF)
    - Nonlinear time-frequency masking
  - Advanced topics
    - Bayesian sound source separation and localization
    - Automatic determination of number of sources

3D Coordinate Systems

- Orthogonal coordinate ↔ Polar coordinate



$$\begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} r\cos\phi\sin\theta \\ r\cos\phi\cos\theta \\ r\sin\phi \end{bmatrix}$$

Used in this lecture

$$\theta = \frac{\pi}{2} - \bar{\theta}$$

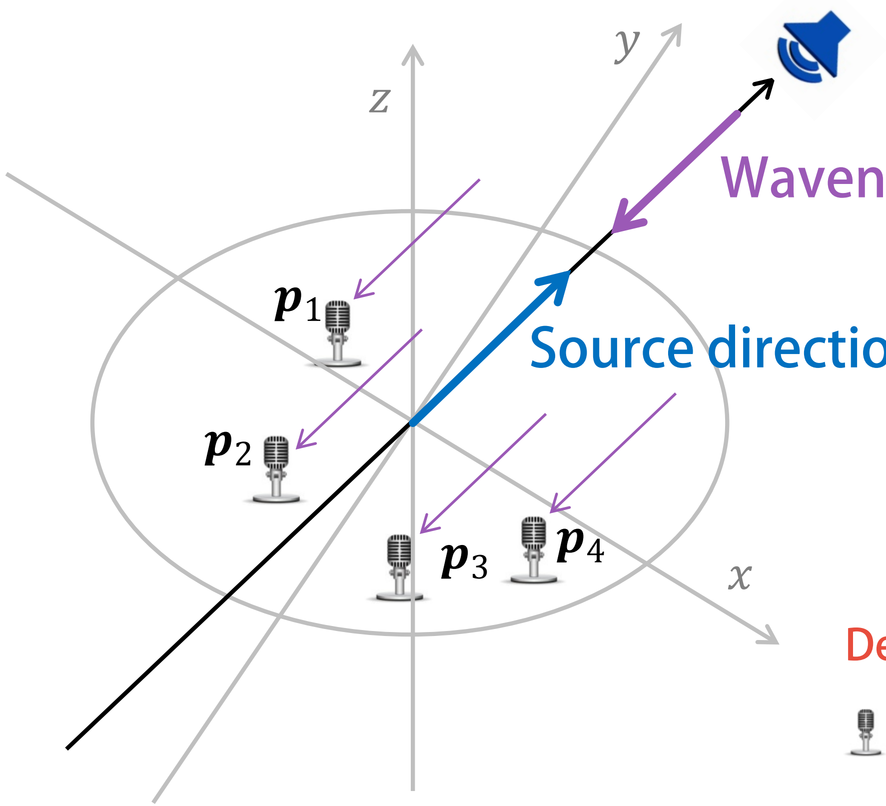
$$\phi = \frac{\pi}{2} - \bar{\phi}$$

$$\begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} r\sin\bar{\phi}\cos\bar{\theta} \\ r\sin\bar{\phi}\sin\bar{\theta} \\ r\cos\bar{\phi} \end{bmatrix}$$

Commonly used in many studies

Propagation of Plain Wave

- A sound wave is observed by using M microphones
  - $\boldsymbol{p}_1, \boldsymbol{p}_2, \cdots, \boldsymbol{p}_M$: the positions of M microhones



ω: angular frequency of plain wave

c: sound speed

Wavenumber vector $\boldsymbol{k} = -\frac{\omega}{c}\boldsymbol{u}$

Source direction $\boldsymbol{u} = \begin{bmatrix} \cos\phi_s \sin\theta_s \\ \cos\phi_s \cos\theta_s \\ \sin\phi_s \end{bmatrix}$

Source Signal → Observed Signals

- An observed signal is a delayed version of a source signal
  - Suppose that source signal $s(t)$ is propagated to M microphones
  - Each microphone m $(1 \leq m \leq M)$ has delay time τ_m

$$\boldsymbol{z}(t) = \begin{bmatrix} z_1(t) \\ z_2(t) \\ \vdots \\ z_M(t) \end{bmatrix} = \begin{bmatrix} s(t-\tau_1) \\ s(t-\tau_2) \\ \vdots \\ s(t-\tau_M) \end{bmatrix} \xrightarrow{\text{Fourier transform}} \boldsymbol{z}(\omega) = \begin{bmatrix} Z_1(\omega) \\ Z_2(\omega) \\ \vdots \\ Z_M(\omega) \end{bmatrix}$$

Observed signals

$$Z_m(\omega) \equiv \int_{-\infty}^{\infty} z_m(t) e^{-j\omega t} dt = \int_{-\infty}^{\infty} s(t-\tau_m) e^{-j\omega t} dt = e^{-j\omega\tau_m} S(\omega)$$

$$S(\omega) \equiv \int_{-\infty}^{\infty} s(t) e^{-j\omega t} dt$$

$$\boldsymbol{a}(\omega) = \begin{bmatrix} a_1 \\ \vdots \\ a_M \end{bmatrix} \equiv \begin{bmatrix} e^{-j\omega\tau_1} \\ \vdots \\ e^{-j\omega\tau_M} \end{bmatrix}$$

Array manifold vector

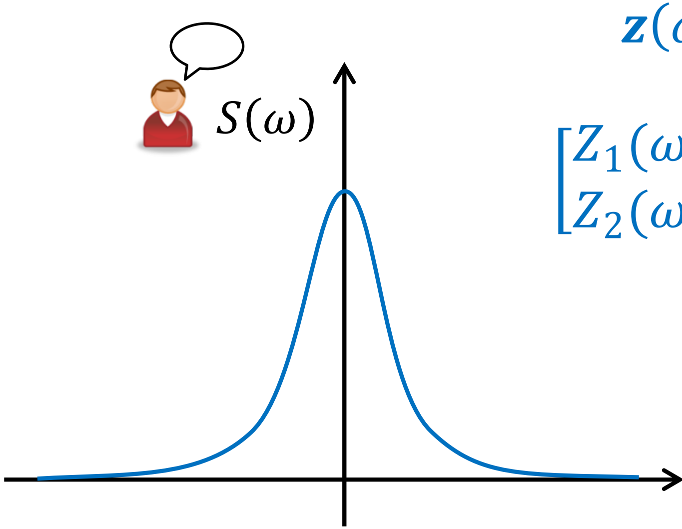
$$\boldsymbol{z}(\omega) = \boldsymbol{a}(\omega) S(\omega)$$

Observation

Source

Sound Observation

- Observed signals are correlated with each other
  - The spatial property is determined by an array manifold vector



We often assume

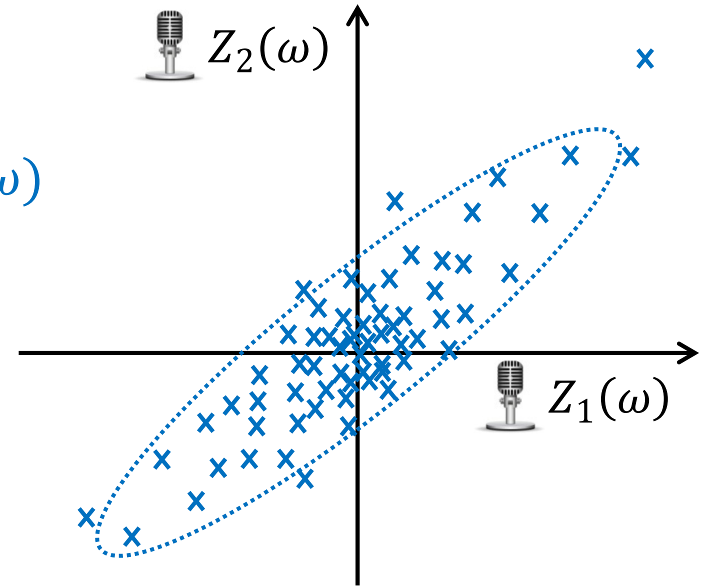
$$S(\omega) \sim N_c(0,1)$$

$$\boldsymbol{z}(\omega) = \boldsymbol{a}(\omega)S(\omega)$$

$$\begin{bmatrix} Z_1(\omega) \\ Z_2(\omega) \end{bmatrix} = \begin{bmatrix} a_1(\omega) \\ a_2(\omega) \end{bmatrix} S(\omega)$$

$$\boldsymbol{a}(\omega) = \begin{bmatrix} a_1 \\ \vdots \\ a_M \end{bmatrix} \equiv \begin{bmatrix} e^{-j\omega\tau_1} \\ \vdots \\ e^{-j\omega\tau_M} \end{bmatrix}$$

Array manifold vector



$$\boldsymbol{z}(\omega) \sim N_c(0, \boldsymbol{a}(\omega)\boldsymbol{a}^H(\omega))$$

Array Manifold Vector for Plane Wave

- The array manifold vector $\boldsymbol{a}(\omega)$ can be calculated from microphone positions $\boldsymbol{p}_1, \boldsymbol{p}_2, \cdots, \boldsymbol{p}_M$

(azimuth, elevation): (θ_s, ϕ_s) Source direction: $\boldsymbol{u} = \begin{bmatrix} \cos\phi_s \sin\theta_s \\ \cos\phi_s \cos\theta_s \\ \sin\phi_s \end{bmatrix}$

Wave equation: $\frac{\partial^2 s}{\partial x^2} + \frac{\partial^2 s}{\partial y^2} + \frac{\partial^2 s}{\partial z^2} = \frac{1}{c^2}\frac{\partial^2 s}{\partial t^2}$

s: sound pressure
c: sound speed

Plain wave with angular frequency ω that solves the equation:

$$s(\boldsymbol{p}, t) = A\exp(j(\omega t - \boldsymbol{k}^T\boldsymbol{p})) = A\exp(j\omega t)\exp(-j\boldsymbol{k}^T\boldsymbol{p})$$

Source signal: $A\exp(j\omega t)$ — Phase difference: $\exp(-j\boldsymbol{k}^T\boldsymbol{p})$

$\boldsymbol{k}$: wavenumber vector

$\boldsymbol{p}$: observation point

λ: wavelength $\quad \lambda = \frac{2\pi c}{\omega} = \frac{c}{f}$

$$\boldsymbol{k} \equiv -\frac{\omega}{c}\boldsymbol{u} = -\frac{2\pi}{\lambda}\boldsymbol{u} \qquad |\boldsymbol{k}| \equiv \frac{\omega}{c}$$

$$a_m(\omega) = \exp(-j\boldsymbol{k}^T\boldsymbol{p}_m) = \exp\left(j\frac{2\pi}{\lambda}\boldsymbol{u}^T\boldsymbol{p}_m\right)$$

$$\tau_m = \frac{1}{\omega}\boldsymbol{k}^T\boldsymbol{p}_m = -\frac{1}{c}\boldsymbol{u}^T\boldsymbol{p}_m$$

Straight-shape Microphone Array

- Microphone position

$$\boldsymbol{p_m} = \left[\left((m-1) - \frac{M-1}{2}\right) d_x, 0, 0\right]^T$$

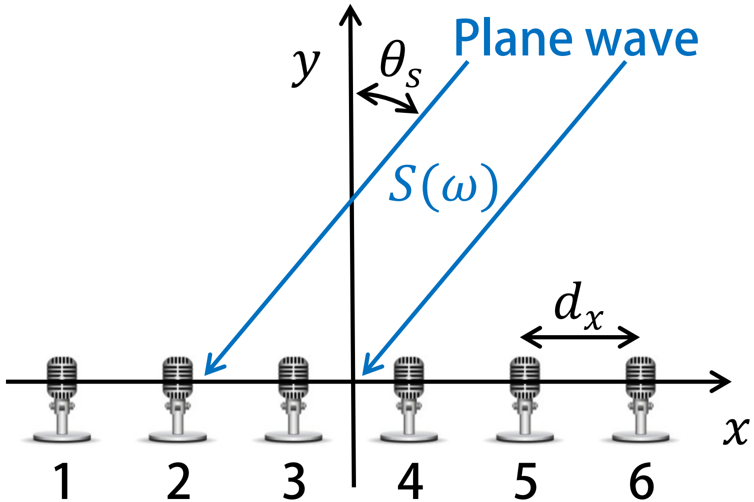
- Time delay

$$\tau_m = -\left((m-1) - \frac{M-1}{2}\right) \frac{d_x}{c} \sin\theta_s$$

- Array manifold vector

$$a_m(\omega) = \exp\left(j\left((m-1) - \frac{M-1}{2}\right) \frac{2\pi d_x}{\lambda} \sin\theta_s\right)$$

$$\boldsymbol{a}(\omega) = e^{-\frac{j(M-1)\psi}{2}}\left[1, e^{j\psi}, e^{j2\psi}, \cdots, e^{j(M-1)\psi}\right]^T \quad \left(\psi = \frac{2\pi d_x}{\lambda} \sin\theta_s\right)$$



$$\boldsymbol{z}(\omega) = [Z_1(\omega), Z_2(\omega), \cdots, Z_6(\omega)]$$

$$\boldsymbol{z}(\omega) = \boldsymbol{a}(\omega) S(\omega)$$

Round-shape Microphone Array

- **Microphone position**

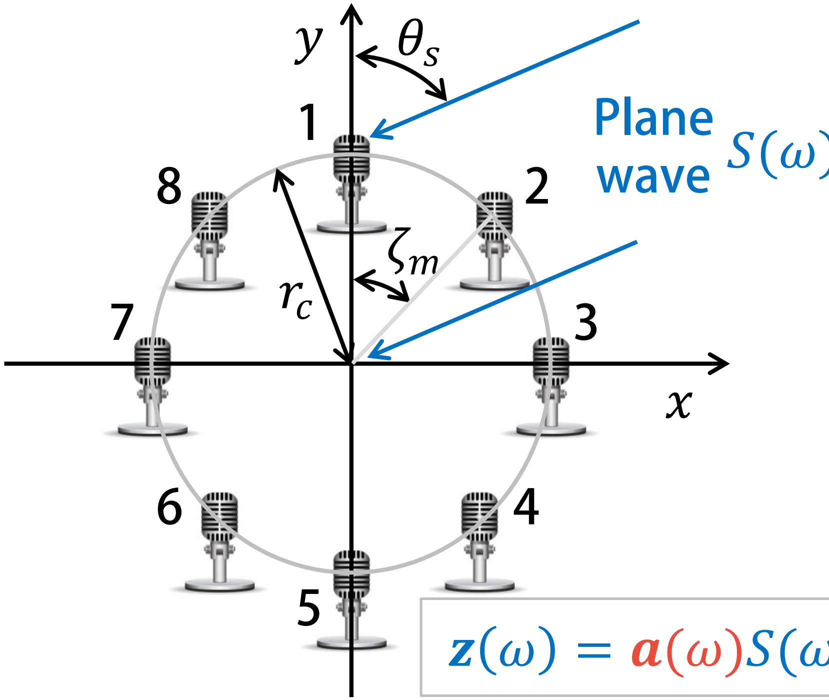
$$\boldsymbol{p}_m = [r_c \sin\zeta_m , r_c \cos\zeta_m , 0]^T$$

- **Time delay**

$$\tau_m = -\frac{r_c}{c}(\sin\theta_s \sin\zeta_m + \cos\theta_s \cos\zeta_m)$$
$$= -\frac{r_c}{c}\cos(\theta_s - \zeta_m)$$

- **Array manifold vector**

$$a_m(\omega) = \exp\left(j\frac{2\pi r_c}{\lambda}\cos(\theta_s - \zeta_m)\right)$$



$$\boldsymbol{z}(\omega) = \boldsymbol{a}(\omega)S(\omega)$$

$$\boldsymbol{z}(\omega) = [Z_1(\omega), Z_2(\omega), \cdots, Z_8(\omega)]$$

Such round-shape microphone arrays are often used in practice for localizing and separating sound sources around a robot

Array Manifold Vector for Spherical Wave

- The array manifold vector $\boldsymbol{a}(\omega)$ can be calculated from 3D microphone positions $\boldsymbol{p}_1, \boldsymbol{p}_2, \cdots, \boldsymbol{p}_M$

(azimuth, elevation, distance): (θ_s, ϕ_s, r) Source position: $\begin{bmatrix} r\cos\phi_s \sin\theta_s \\ r\cos\phi_s\cos\theta_s \\ r\sin\phi_s \end{bmatrix}$

Wave equation: $$\frac{\partial^2(rs)}{\partial r^2} = \frac{1}{c^2}\frac{\partial^2(rs)}{\partial t^2}$$

s: sound pressure
c: sound speed

Spherical wave with angular frequency ω that satisfies the equation:

$$s(r,t) = \frac{A}{r}\exp(j(\omega t - k_r r)) = A\exp(j\omega t)\frac{1}{r}\exp(-jk_r r)$$

Source signal ($A\exp(j\omega t)$)
Phase/amplitude difference ($\frac{1}{r}\exp(-jk_r r)$)

k_r: wavenumber
λ: wavelength

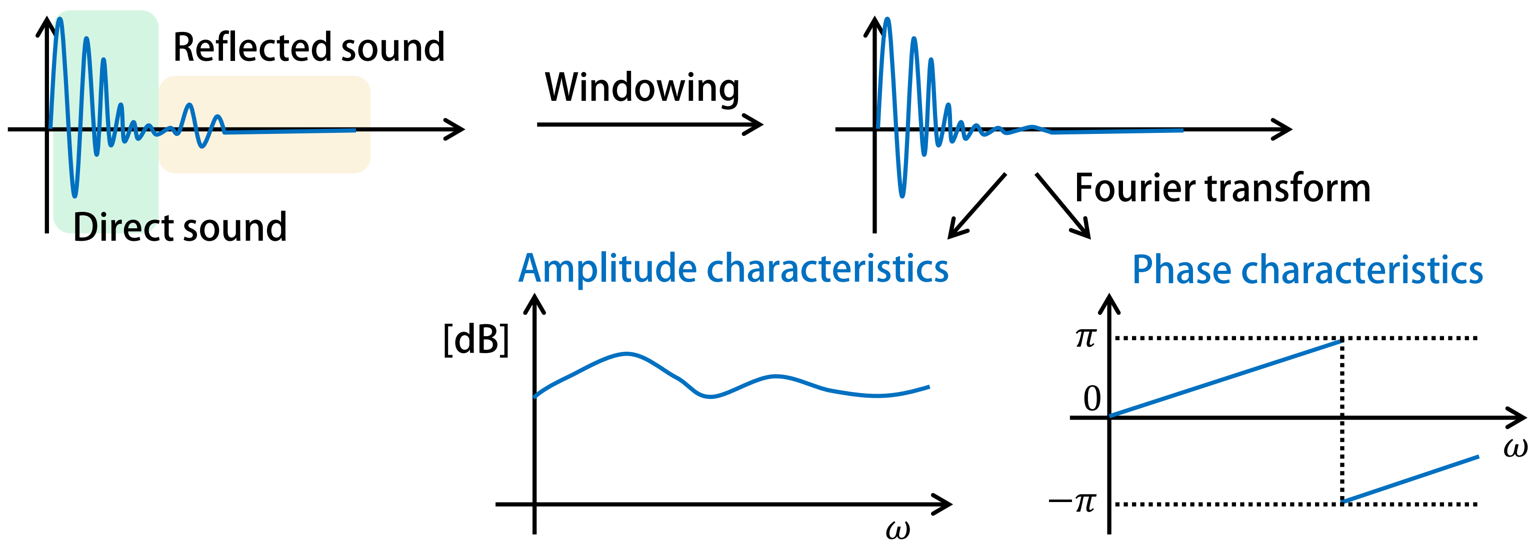
$$k_r \equiv \frac{2\pi}{\lambda} = \frac{\omega}{c} \qquad \lambda = \frac{2\pi c}{\omega} = \frac{c}{f}$$

$$a_m(\omega) = \frac{1}{r_m}\exp(-jk_r r_m) = \frac{1}{r_m}\exp\left(-j\omega\frac{r_m}{c}\right)$$

Measurement of Array Manifold Vector

- Geometry-based estimation
  - Use the formula: $a_m(\omega) = \exp(-j\boldsymbol{k}^T\boldsymbol{p}) = \exp\left(j\frac{2\pi}{\lambda}\boldsymbol{u}^T\boldsymbol{p}_m\right)$

    Source direction ($\boldsymbol{u}$) / Microphone position ($\boldsymbol{p}_m$)
- Recording-based estimation
  - Use only direct sounds for measuring the impulse response
  - Transform the impulse response into the frequency domain



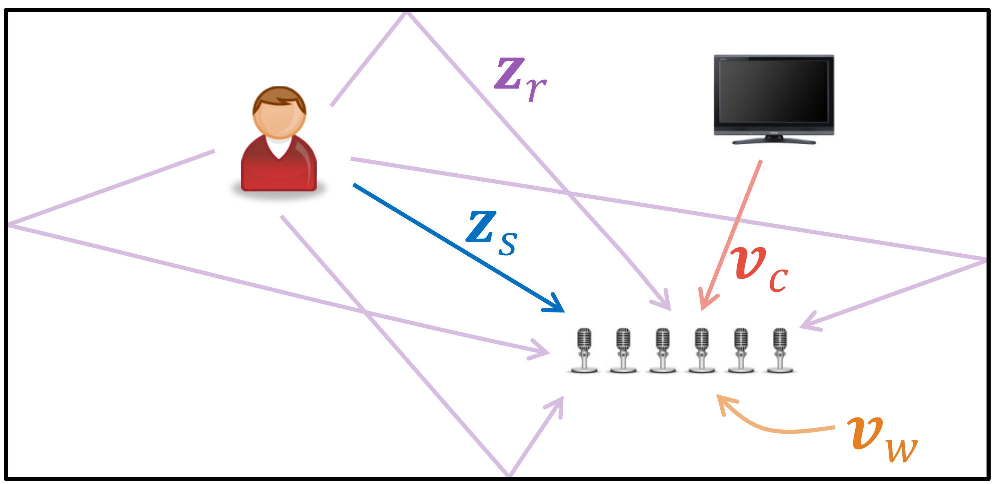
Sound Observation Model

- The observed sound is a mixture of various sounds
  - Direct sound: z_s
  - Reflected sound: z_r
  - Spatial colored noise: v_c
  - Spatial white noise: v_w

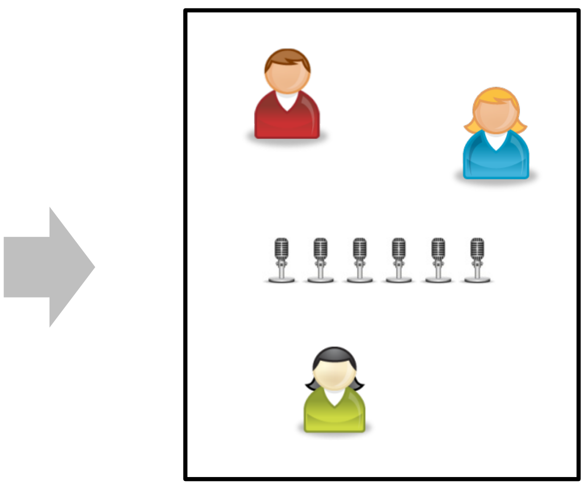
Observed sound:

$$\boldsymbol{z} = \boldsymbol{z}_s + \boldsymbol{z}_r + \boldsymbol{v}_c + \boldsymbol{v}_w$$

Single source

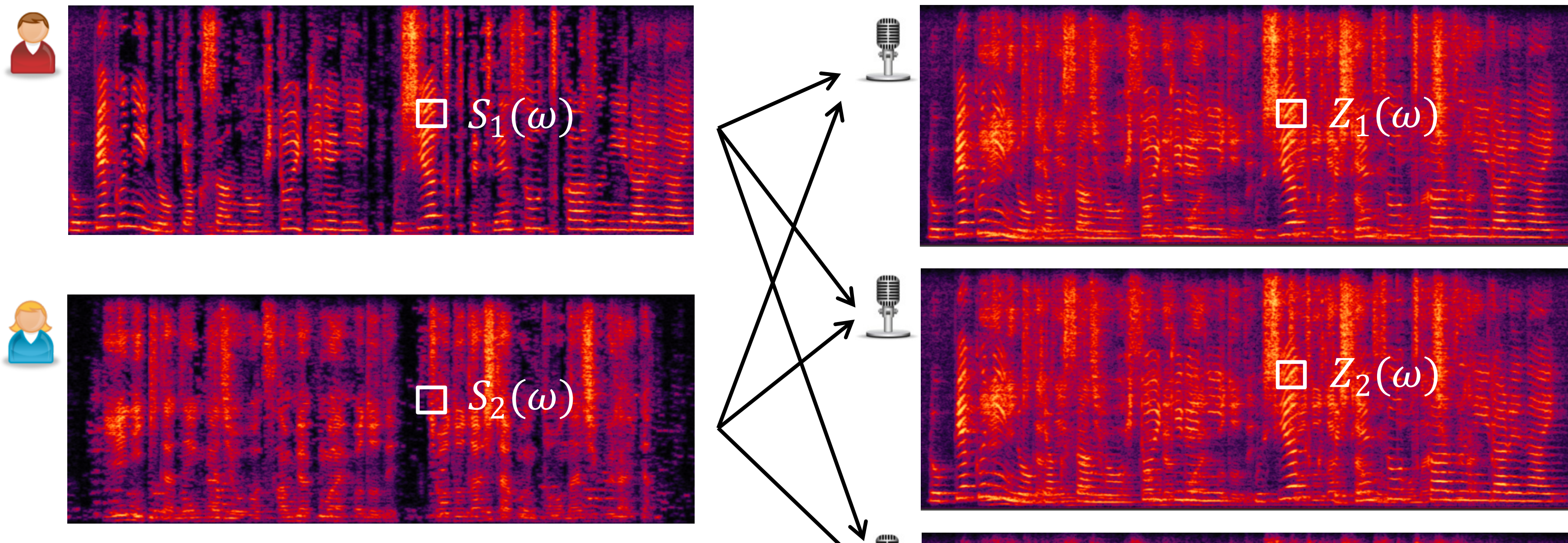


Multiple sources



Sound Observation Model

- Suppose that N sound sources and M microphones



$$\boldsymbol{s}(\omega) = \begin{bmatrix} S_1(\omega) \\ S_2(\omega) \\ \vdots \\ S_M(\omega) \end{bmatrix} \quad \boldsymbol{z}(\omega) = \begin{bmatrix} Z_1(\omega) \\ Z_2(\omega) \\ \vdots \\ Z_N(\omega) \end{bmatrix}$$

Direct Sounds

- Sum of direct sounds coming from N sound sources
  - Suppose that there are N sound sources
  - Each sound source is recorded by each microphone (linear system)

Single source

$$\boldsymbol{z}_s(\omega) = \boldsymbol{a}(\omega)S(\omega)$$

→

Multiple sources

$$\boldsymbol{z}_s(\omega) = \sum_{i=1}^{N} \boldsymbol{a}_i(\omega)S_i(\omega) = \boldsymbol{A}(\omega)\boldsymbol{s}(\omega)$$

$$\boldsymbol{z}_s(\omega) = \begin{bmatrix} Z_{s1}(\omega) \\ Z_{s2}(\omega) \\ \vdots \\ Z_{sM}(\omega) \end{bmatrix}$$

Mixing matrix
(array manifold matrix)

$$\boldsymbol{A}(\omega) = [\boldsymbol{a}_1(\omega), \cdots, \boldsymbol{a}_N(\omega)]$$

$\boldsymbol{a}_n(\omega)$: array manifold vector for each source n

$$\boldsymbol{s}(\omega) = \begin{bmatrix} S_1(\omega) \\ S_2(\omega) \\ \vdots \\ S_N(\omega) \end{bmatrix}$$

Reflected Sounds and Noise

- Different linear systems are assumed
  - Direct sounds: $\boldsymbol{z}_s = \boldsymbol{A}\boldsymbol{s}$
  - Reflected sounds: $\boldsymbol{z}_r = \boldsymbol{A}_r\check{\boldsymbol{s}}$ ($\check{\boldsymbol{s}}$ is highly correlated to $\boldsymbol{s}$)
    - Short direct path ≠ Long reflection path → $\boldsymbol{A} \neq \boldsymbol{A}_r$
  - Spatial colored noise: $\boldsymbol{v}_c = \boldsymbol{A}_c\boldsymbol{q}$ ($\boldsymbol{q}$ is not correlated to $\boldsymbol{s}$)
    - The elements of $\boldsymbol{v}_c$ are inter-dependent
  - Spatial white noise: $\boldsymbol{v}_w \sim N(0, \sigma^2\boldsymbol{I})$
    - The elements of $\boldsymbol{v}_w$ are independent

$$\boldsymbol{v} = \boldsymbol{v}_c + \boldsymbol{v}_w$$

$\boldsymbol{v}_c$ is often assumed to be included in $\boldsymbol{v}_w$

General observation model

$$\boldsymbol{z} = \boldsymbol{A}\boldsymbol{s} + \boldsymbol{v}$$

Spatial Correlation Matrix

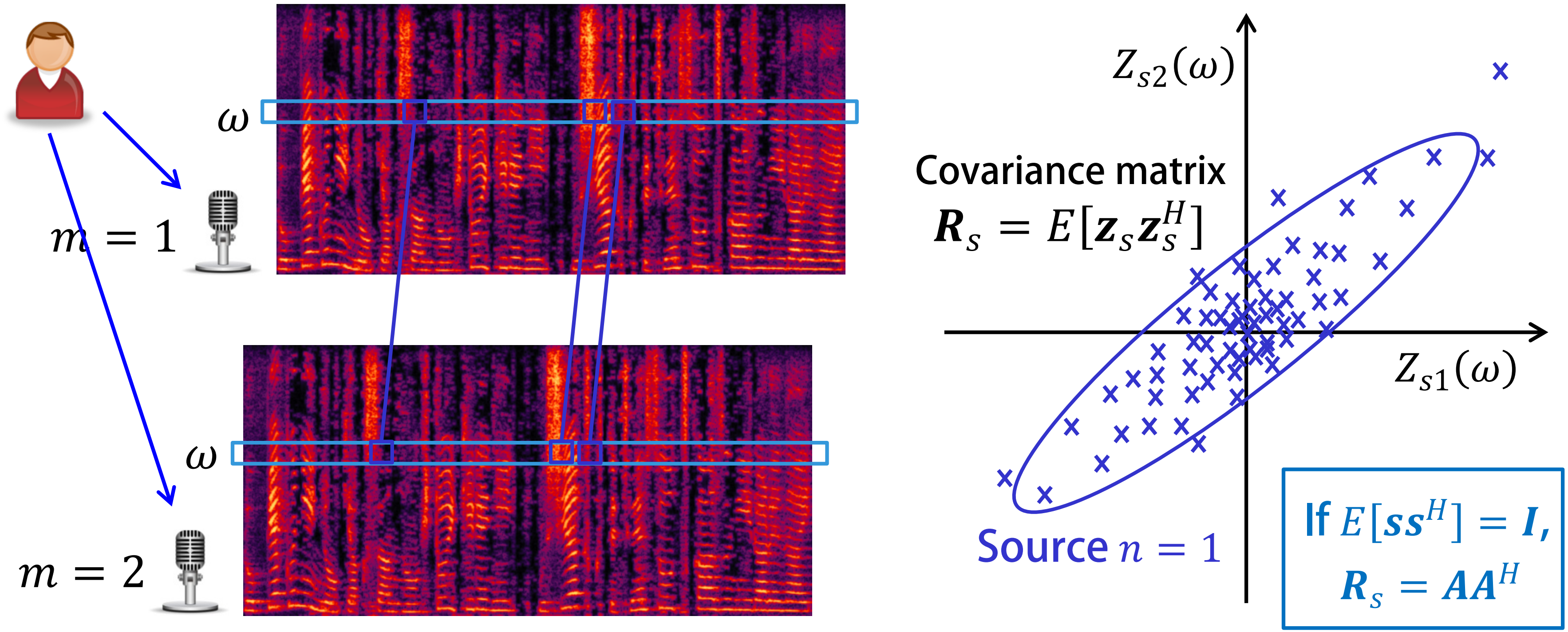
- The spatial correlation matrix $\boldsymbol{R} = E[\boldsymbol{z}\boldsymbol{z}^H]$ represents the spatial characteristics of multi-channel signals $\boldsymbol{z}$
  - For direct sounds: $\boldsymbol{R}_s = E[\boldsymbol{z}_s\boldsymbol{z}_s^H] = \boldsymbol{A}E[\boldsymbol{s}\boldsymbol{s}^H]\boldsymbol{A}^H = \boldsymbol{A}\boldsymbol{\Gamma}\boldsymbol{A}^H$
  - For source signals: $\boldsymbol{\Gamma} = \mathrm{E}[\boldsymbol{s}\boldsymbol{s}^H]$
    - If sound signals are independent, $\boldsymbol{\Gamma} = \mathrm{diag}(\gamma_1, \cdots, \gamma_N)$
    - $\gamma_i = E[S_i(\omega)S_i^*(\omega)]$ is the power of source i at frequency ω
  - For noise: $\boldsymbol{K} = E[\boldsymbol{v}\boldsymbol{v}^H]$
    - If noise $\boldsymbol{v}$ is spatially white, $\boldsymbol{K} = \sigma^2\boldsymbol{I}$
    - σ^2 is the power of noise

General observation model: $\boldsymbol{z} = \boldsymbol{A}\boldsymbol{s} + \boldsymbol{v}$

Observation of Single Source

- The spectra of each source has a unique spatial property
  - The spatial correlation matrix $\boldsymbol{R}_s$ is determined by the mixing matrix $\boldsymbol{A}$

Observed data: $\boldsymbol{z}_s(\omega) = [Z_{s1}(\omega), \cdots, Z_{sM}(\omega)]$



Probabilistic Modeling

- Formulate a probabilistic model of $\boldsymbol{z} = \boldsymbol{A}\boldsymbol{s} + \boldsymbol{v}$
  - Deterministic signal model

$$p(\boldsymbol{v}) = N(\boldsymbol{v}|\boldsymbol{0}, \boldsymbol{K}) \xrightarrow[\boldsymbol{z} = \boldsymbol{A}\boldsymbol{s} + \boldsymbol{v}]{\text{Linear transform}}$$

Likelihood: $p(\boldsymbol{z}; \boldsymbol{\Theta}) = N(\boldsymbol{z}|\boldsymbol{A}\boldsymbol{s}, \boldsymbol{K})$

Find $\boldsymbol{\Theta} = \{\boldsymbol{A}, \boldsymbol{s}, \boldsymbol{K}\}$ that maximizes $p(\boldsymbol{z}; \boldsymbol{\Theta})$

  - Random signal model

$\boldsymbol{A}$ is determined by source directions $\{\theta_1, \cdots, \theta_N\}$
$\boldsymbol{\Gamma}$ is determined by source power $\{\gamma_1, \cdots, \gamma_N\}$

$$p(\boldsymbol{v}) = N(\boldsymbol{v}|\boldsymbol{0}, \boldsymbol{K})$$
$$p(\boldsymbol{s}) = N(\boldsymbol{s}|\boldsymbol{0}, \boldsymbol{\Gamma})$$

Linear transform
$$\boldsymbol{z} = \boldsymbol{A}\boldsymbol{s} + \boldsymbol{v}$$
$$\boldsymbol{\Gamma} = E[\boldsymbol{s}\boldsymbol{s}^H] \ (= \mathrm{diag}(\gamma_1, \cdots, \gamma_N))$$

Likelihood: $p(\boldsymbol{z}; \boldsymbol{\Theta}) = N\left(\boldsymbol{z}\middle|\boldsymbol{0}, \boldsymbol{A}\boldsymbol{\Gamma}\boldsymbol{A}^H + \boldsymbol{K}\right)$

Find $\boldsymbol{\Theta} = \{\boldsymbol{A}, \boldsymbol{\Gamma}, \boldsymbol{K}\}$ that maximizes $p(\boldsymbol{z}; \boldsymbol{\Theta})$

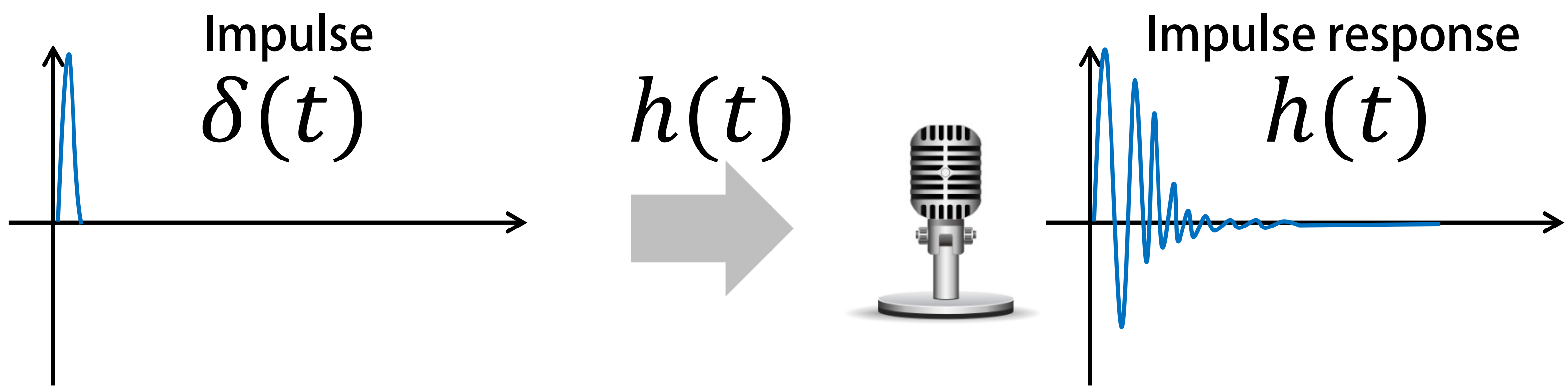
Bayesian treatment of $\boldsymbol{\Theta}$ is feasible by incorporating a prior $p(\boldsymbol{\Theta})$

$$p(\boldsymbol{\Theta}|\boldsymbol{z}) = \frac{p(\boldsymbol{z}|\boldsymbol{\Theta})p(\boldsymbol{\Theta})}{p(z)}$$

Impulse Response

Impulse Response

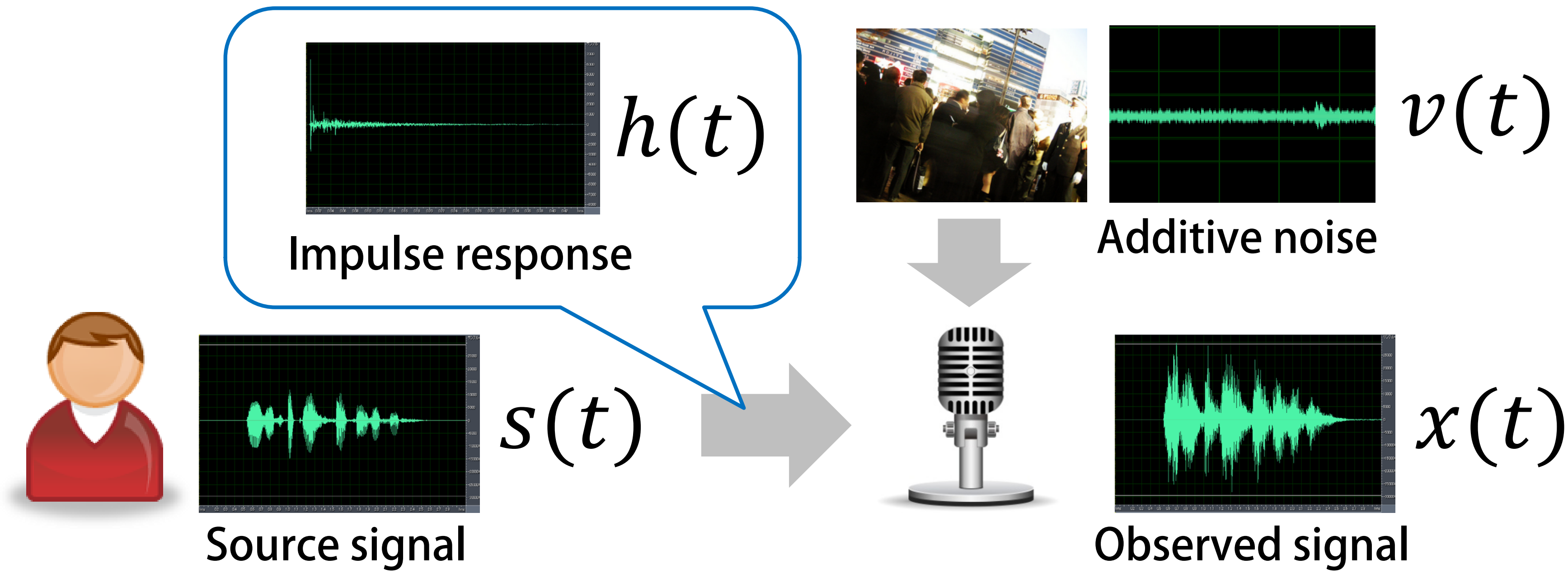
- The impulse response is a signal recorded by a microphone when an impulse is emitted from a sound source
  - The source signal is distorted by reflection, noise, and diffraction
  - Impulse response (time domain) = Transfer function (freq. domain)
    - Different rooms have different impulse responses



$$h(t) = h(t) * \delta(t)$$

Basic Formulation

- Room acoustics are often represented as a linear system
  - Source signal + Room acoustics + Additive noise → Observed signal



$$z(t) = h(t) * s(t) + v(t)$$

Convolution of Impulse Response

- Time-domain convolution ↔ Frequency-domain product
  - $s(t)$: source signal → $z(t)$: observed signal
  - $h(t)$: impulse response that characterizes the linear system

Continuous time domain

$$z(t) = h(t) * s(t) = \int_{-\infty}^{\infty} h(t-\tau)s(\tau)d\tau$$

Discrete time domain

$$z[t] = h[t] * s[t] = \sum_{\tau=-\infty}^{\infty} h[t-\tau]s[\tau]$$

n: time index

Time domain

$$z[n] = h[n] * s[n] = \sum_{m=0}^{N-1} h[n-m]s[m]$$

Freq. domain

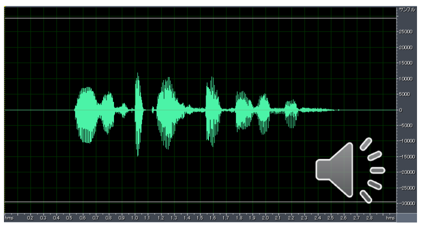
$$Z[k] = H[k] \cdot S[k]$$

k: frequency index

Simulation of Room Acoustics

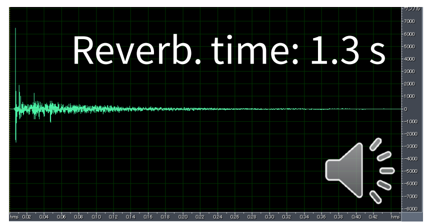
- Audio signals recorded in an arbitrary room can be simulated by using the impulse response of the room

Source signal



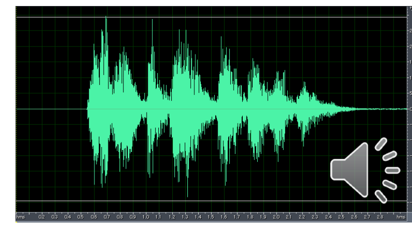
$*$

Impulse response



$=$

Observed signal



Time-domain representation

$$s(t) * h(t) = z(t)$$

FFT ↓   ↑ iFFT

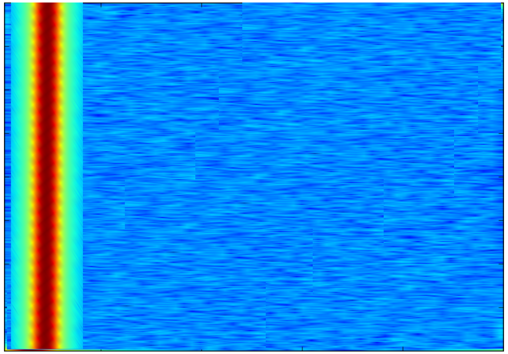
Frequency-domain representation

$$S(\omega) \times H(\omega) = Z(\omega)$$

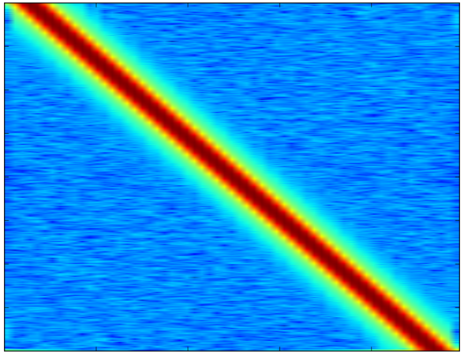
Time Stretched Pulse

- The TSP can be easily emitted from a loudspeaker
  - Frequency characteristics:
    - The impulse contains all frequencies at a moment (huge power)
    - The TSP contains a limited range of frequencies at a moment
  - The impulse is recovered by convoluting two TSPs

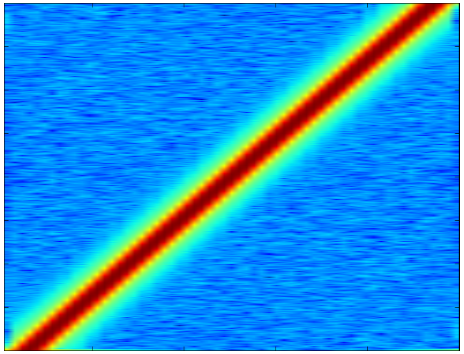
Impulse



Time stretched pulse



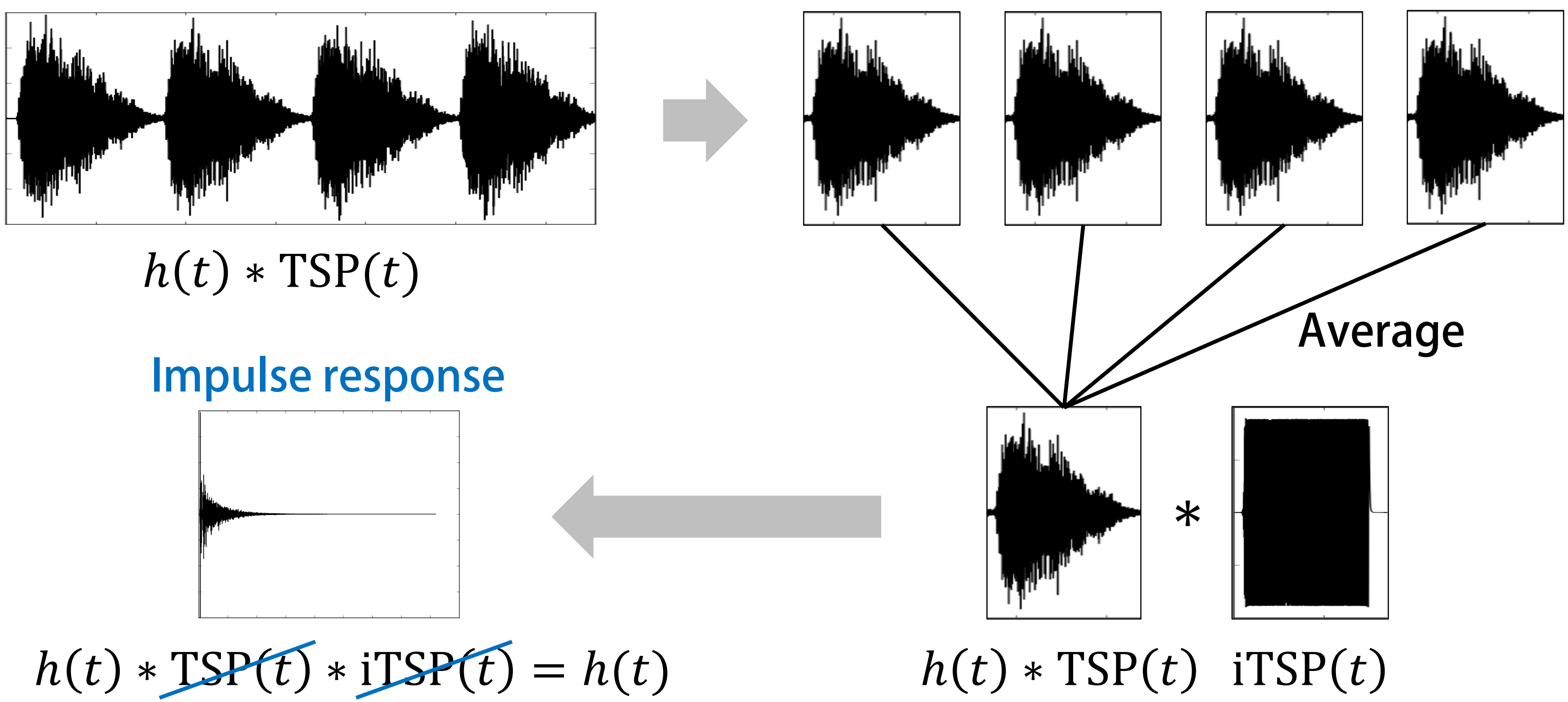
Inverse TSP



$$\delta(t) = \mathrm{TSP}(t) * \mathrm{iTSP}(t)$$

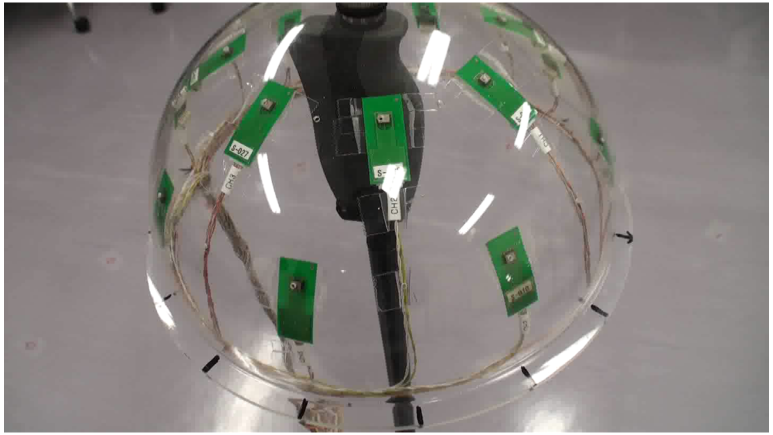
Measurement of Impulse Response

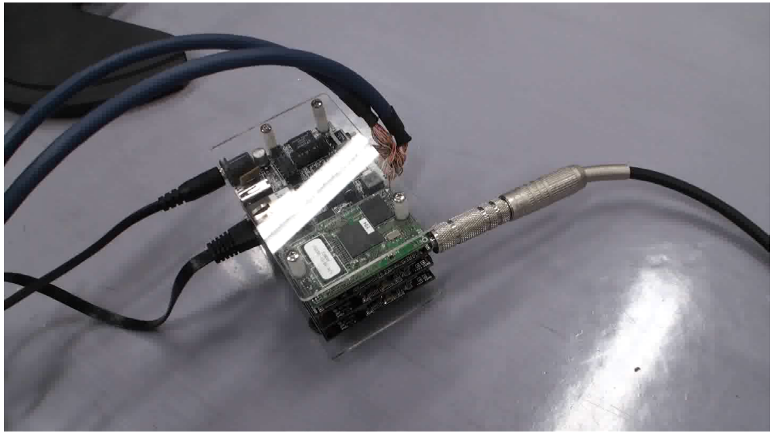
- Convolute a TSP response with an inverse TSP
  - The effects of TSP and iTSP are canceled out



Recording Setting

- Prepare devices required for recording TSPs





Loudspeaker and earplugs:
The TSPs are emitted multiple times

Microphone array:
The TSP is recorded by each microphone

Recording device:
All microphones are synchronized

Recording Setting

- Mark the floor with a certain interval (5° or 10°)

Angle measurer:
  Laser is emitted while rotating

Markers:
  Stickers are on all directions

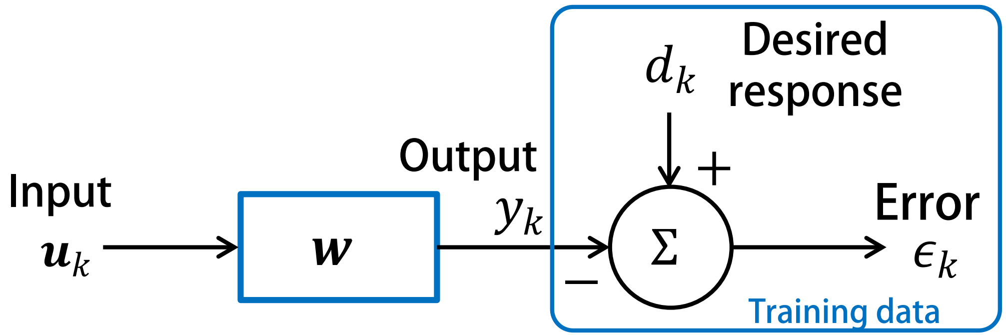
Two people:
  Angle measurer control + Marking

TSP Recording

Wiener Filtering

Wiener Filter

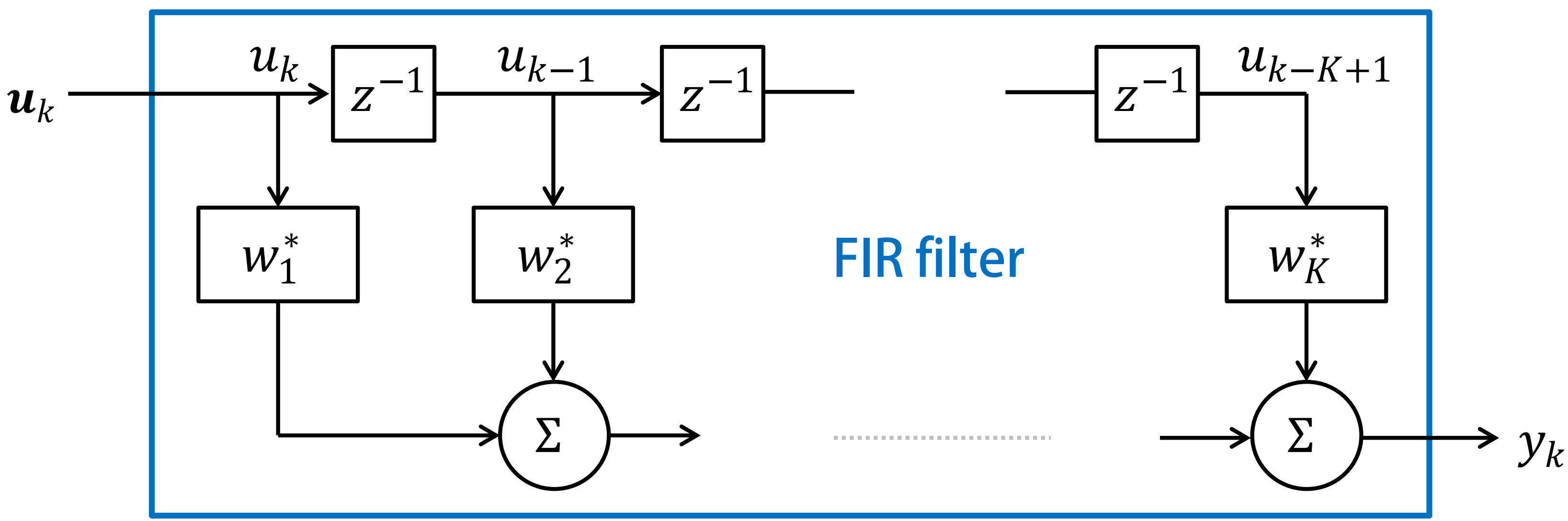
- We aim to learn a linear filter that extracts signals of interest



$$y_k = \boldsymbol{w}^H \boldsymbol{u}_k$$

k: time index

$$\boldsymbol{w} = \begin{bmatrix} w_1 \\ w_2 \\ \vdots \\ w_K \end{bmatrix} \quad \boldsymbol{u}_k = \begin{bmatrix} u_k \\ u_{k-1} \\ \vdots \\ u_{k-K+1} \end{bmatrix}$$



Supervised Learning

- Estimate a linear filter $\boldsymbol{w}_{\mathrm{MF}}$ that extracts y_k from an observed signal $\boldsymbol{u}_k$ such that y_k is close to a given desired response d_k
  - Minimize error ϵ_k between desired response d_k and filter output y_k

  Cost function

$$
\begin{aligned}
J &= E\left[|\epsilon_k|^2\right] \\
&= E\left[\left(d_k - \boldsymbol{w}^H \boldsymbol{u}_k\right)\left(d_k - \boldsymbol{w}^H \boldsymbol{u}_k\right)^H\right] \\
&= E[d_k d_k^*] - \boldsymbol{w}^H E[\boldsymbol{u}_k d_k^*] - E\left[d_k^* \boldsymbol{u}_k^H\right]\boldsymbol{w} + \boldsymbol{w}^H E\left[\boldsymbol{u}_k \boldsymbol{u}_k^H\right]\boldsymbol{w} \\
&\equiv \boldsymbol{\sigma}_d^2 - \boldsymbol{w}^H \boldsymbol{r}_{ud} - \boldsymbol{r}_{ud}^H \boldsymbol{w} + \boldsymbol{w}^H \boldsymbol{R}_u \boldsymbol{w}
\end{aligned}
$$

  Let the partial derivative be equal to zero

$$
\frac{\partial J}{\partial \boldsymbol{w}^*} = -\boldsymbol{r}_{ud} + \boldsymbol{R}_u \boldsymbol{w} \to 0 \quad \Rightarrow \quad \boldsymbol{R}_u \boldsymbol{w}_{\mathrm{MF}} = \boldsymbol{r}_{ud} \quad \Rightarrow \quad \boldsymbol{w}_{\mathrm{MF}} = \boldsymbol{R}_u^{-1} \boldsymbol{r}_{ud}
$$

Normal equation

Time-domain Representation

- **Wiener filter assumes that input $\boldsymbol{u}_k$ is weakly stationary**
  - The mean and autocorrelation of $\boldsymbol{u}_k$ are constant for any k
  - Auto-correlation: $r_u(n) = E[u_k u_{k-n}^*]$
  - Cross-correlation: $r_{du}(n) = E[d_k u_{k-n}^*]$

Correlation matrix (Toeplitz matrix)

$$\boldsymbol{R}_u = \begin{bmatrix} r_u(0) & \cdots & r_u(K-1) \\ r_u(-1) & & r_u(K-2) \\ \vdots & \ddots & \vdots \\ r_u(1-K) & \cdots & r_u(0) \end{bmatrix} \qquad \boldsymbol{r}_{du} = \begin{bmatrix} r_{du}(0) \\ r_{du}(-1) \\ \vdots \\ r_{du}(K-1) \end{bmatrix}$$

Time-domain Wiener filter

$$\boldsymbol{w}_{\mathrm{MF}} = \boldsymbol{R}_u^{-1} \boldsymbol{r}_{ud} \;\Rightarrow\; \boldsymbol{w}_{\mathrm{MF}}^H \boldsymbol{R}_u = \boldsymbol{r}_{du} \;\Rightarrow\; \sum_{i=1}^{K} w_i^* r_u(n-i+1) = r_{du}(n) \quad (n = 0, \cdots, K-1)$$

Frequency-domain Representation

- **Wiener filter assumes that input $\boldsymbol{u}_k$ is weakly stationary**
  - The mean and autocorrelation of $\boldsymbol{u}_k$ are constant for any k
  - Auto-correlation: $r_u(n) = E[u_k u_{k-n}^*]$
  - Cross-correlation: $r_{du}(n) = E[d_k u_{k-n}^*]$

$$\sum_{i=1}^{K} w_i^* r_u(n-i+1) = r_{du}(n) \quad \Rightarrow \quad \sum_{i=-\infty}^{\infty} w_i^* r_u(n-i+1) = r_{du}(n)$$

We assume that u_k (input) $= d_k$ (desired response) $+ v_k$ (noise)

$$W(\omega)S_u(\omega) = S_{du}(\omega)$$

$$S_u(\omega) = S_d(\omega) + S_v(\omega)$$

$$S_{du}(\omega) = S_d(\omega)$$

d_k and v_k are independent

Frequency-domain Wiener filer

$$W(\omega) = \frac{S_d(\omega)}{S_d(\omega) + S_v(\omega)}$$

Learning Methods

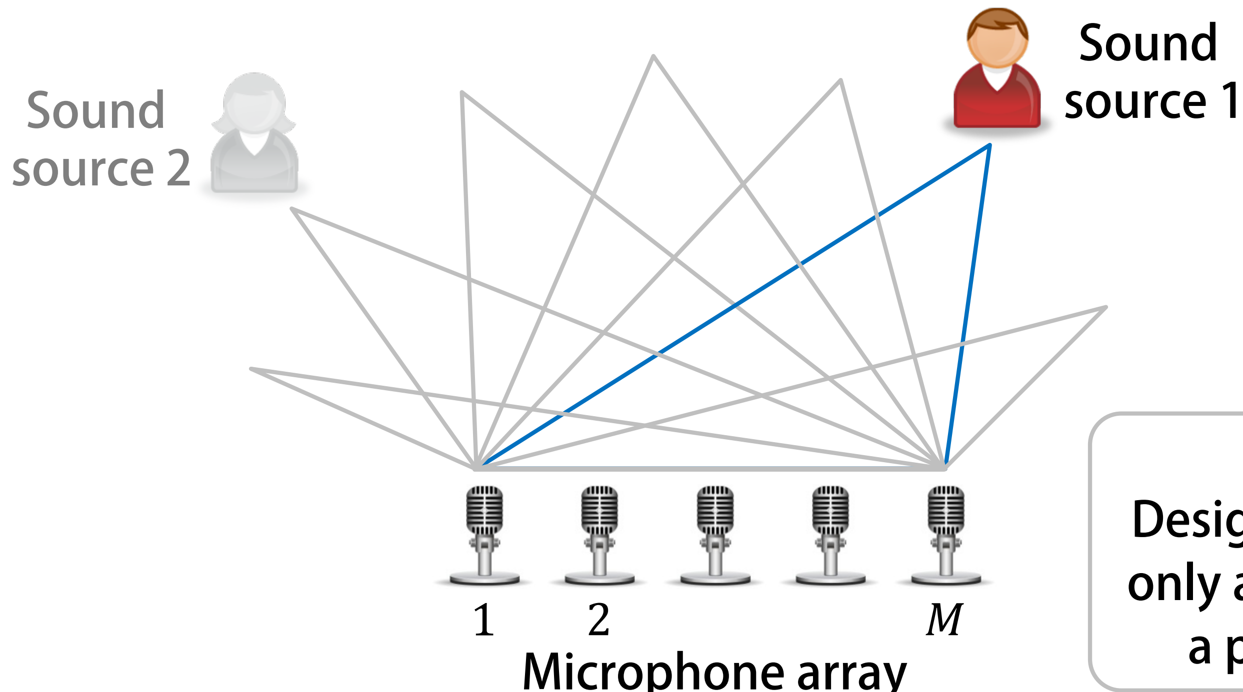
- Fixed filtering
  - Estimate a Wiener filter from a finite amount of samples
    - Least square method (LS)
- Adaptive filtering
  - Estimate a Wiener filter in an online manner
    - Steepest descent method
    - Newton's method
    - Least mean square method (LMS)
    - Affine projection algorithm (APA)
    - Recursive least squares method (RLS)

Beamforming

Beamformer

- Extract signals of a particular direction from observations
  - Assumption: array manifold vectors $\boldsymbol{a}_1, \cdots \boldsymbol{a}_M$ are known

Depend on direction θ, ϕ and frequency ω

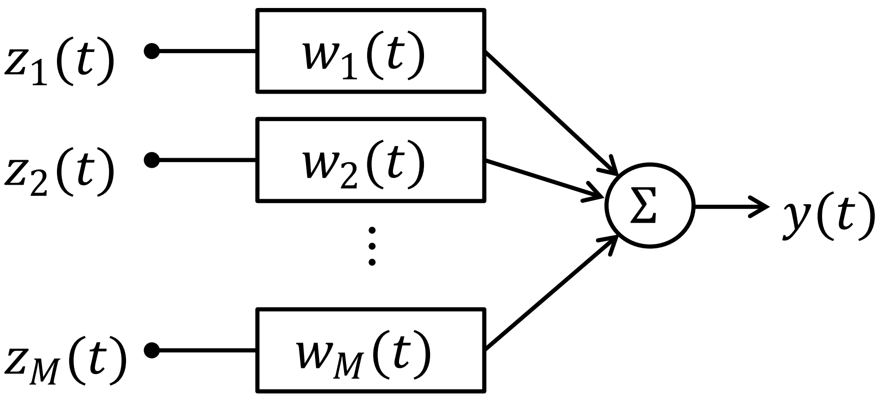


Goal
Design a filter that passes only a signal coming from a particular direction

Basic Formulation

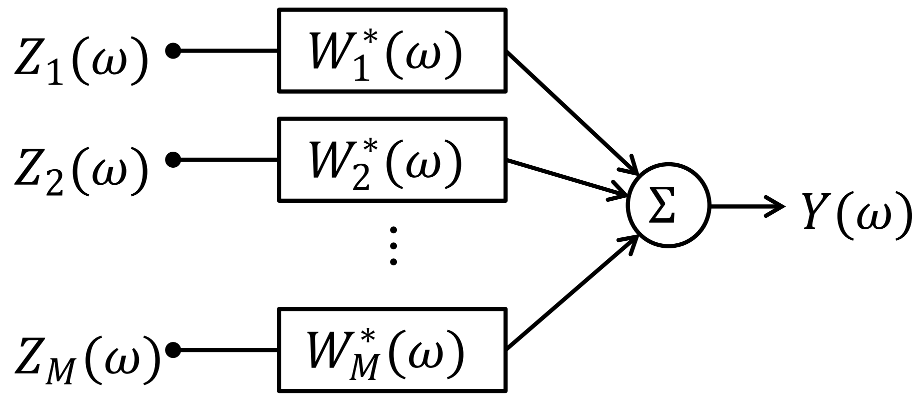
- Design filters passing signals of a particular direction
  - $z_m(t)$: m^{th} observed signal $w_m(t)$: m^{th} filter
  - $y(t)$: output of beamformer

Time domain



$$y(t) = \sum_{m=1}^{M} w_m(t) * z_m(t)$$

Frequency domain

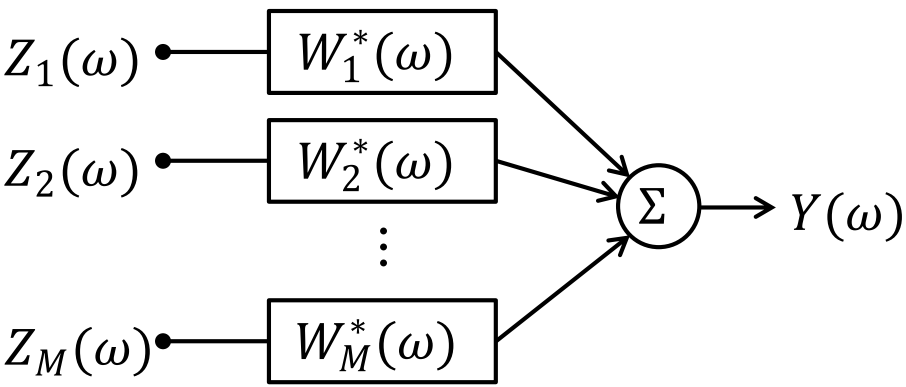


$$Y(\omega) = \sum_{m=1}^{M} W_m^*(\omega) Z_m(\omega)$$

Basic Formulation

- Design filters passing components of a particular direction
  - $Z_m(\omega)$: m^{th} observed signal $W_m(\omega)$: m^{th} filter
  - $Y(\omega)$: output of beamformer

Frequency domain



$$Y(\omega) = \sum_{m=1}^{M} W_m^*(\omega) Z_m(\omega)$$

Vectorial representation

$$\boldsymbol{z}(\omega) = \begin{bmatrix} Z_1(\omega) \\ Z_2(\omega) \\ \vdots \\ Z_M(\omega) \end{bmatrix} \quad \boldsymbol{w}(\omega) = \begin{bmatrix} W_1(\omega) \\ W_2(\omega) \\ \vdots \\ W_M(\omega) \end{bmatrix}$$

$$Y(\omega) = \boldsymbol{w}^H(\omega)\boldsymbol{z}(\omega)$$

Estimated source ($Y(\omega)$)

Observation ($\boldsymbol{z}(\omega)$)

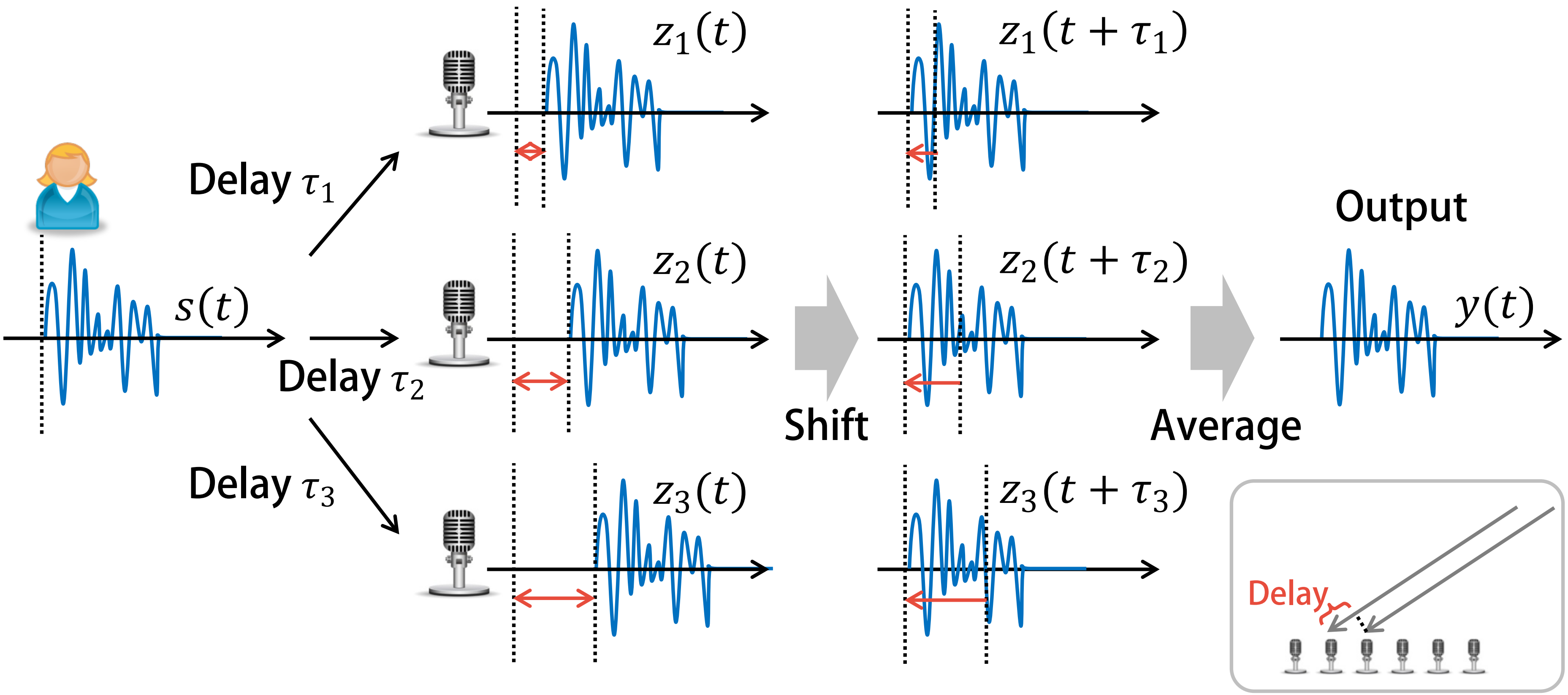
Filter Estimation

- Various methods have been proposed for filter estimation

Method	Filter vector (steering vector)	Beam/filter type and assumptions
Delay-sum beamformer (DS)	$\boldsymbol{w} = \frac{\boldsymbol{a}}{\boldsymbol{a}^H \boldsymbol{a}}$	Beam (fixed filter) $\boldsymbol{a}$: known
Spatial Wiener filter (SWF)	$\boldsymbol{w} = \boldsymbol{R}_Z^{-1} \boldsymbol{r}_{zd}$	Beam & null (adaptive filter) $\boldsymbol{d}$: known
Maximum likelihood (ML)	$\boldsymbol{w} = \frac{\boldsymbol{K}^{-1}\boldsymbol{a}}{\boldsymbol{a}^H \boldsymbol{K}^{-1} \boldsymbol{a}}$	Beam & null (adaptive filter) $\boldsymbol{a}, \boldsymbol{K}$: known
Minimum variance (MV)	$\boldsymbol{w} = \frac{\boldsymbol{R}^{-1}\boldsymbol{a}}{\boldsymbol{a}^H \boldsymbol{R}^{-1} \boldsymbol{a}}$	Beam & null (adaptive filter) $\boldsymbol{a}$: known
Generalized sidelobe canceller (GSC)	$\boldsymbol{w} = (\boldsymbol{B}^H \boldsymbol{R} \boldsymbol{B})^{-1} \boldsymbol{B}^H \boldsymbol{R} \boldsymbol{w}_c$	Beam (fixed) & null (adaptive) $\boldsymbol{a}, \boldsymbol{K}$: known
Generalized eigenvalue decomposition (GEVD)	$\boldsymbol{w} = \boldsymbol{E}\boldsymbol{G}\boldsymbol{E}^{-1}$	Beam & null (adaptive filter) $\boldsymbol{K}$: known

Delay-Sum Beamformer

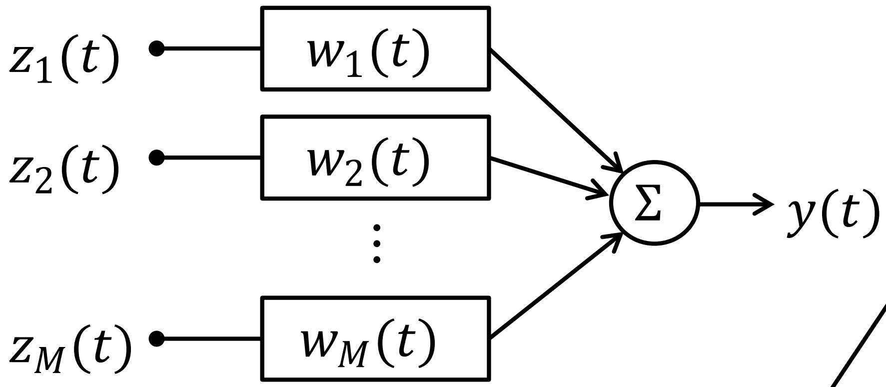
- Take the average of delay-compensated observed signals
  - The delays are determined by a direction of beamforming



Mathematical Formulation

- The filter vector $\boldsymbol{w}$ has a same direction as $\boldsymbol{a}$

Observed signal | Time-domain filter | Source signal of a particular direction



$$w_m(t) = \frac{1}{M}\delta(t + \tau_m)$$

$$y(t) = \sum_{m=1}^{M} w_m(t) * z_m(t)$$

$$= \frac{1}{M}\sum_{m=1}^{M} z_m(t + \tau_m)$$

DS beamformer

Fourier transform

Steering vector $\boldsymbol{w}^H(\omega) = \frac{1}{M}[e^{j\omega\tau_1}, e^{j\omega\tau_2}, \cdots, e^{j\omega\tau_M}]$

Array manifold vector $\boldsymbol{a}(\omega) = \begin{bmatrix} e^{-j\omega\tau_1} \\ \vdots \\ e^{-j\omega\tau_M} \end{bmatrix}$

$$\boldsymbol{w}_{\mathrm{DS}} = \frac{1}{M}\boldsymbol{a} = \frac{\boldsymbol{a}}{\boldsymbol{a}^H\boldsymbol{a}}$$

Normalize

Performance Analysis

- Suppose that a beam with a "wrong" direction is used
  - Source direction (θ_s, ϕ_s)
  - Steering-vector direction (θ_T, ϕ_T)

  Different!

$$\boldsymbol{z}(\omega) = \boldsymbol{a}(\omega)S(\omega)$$

Observation — Source

$$Y(\omega) = \boldsymbol{w}^H(\omega)\boldsymbol{z}(\omega)$$

Estimated source — Observation

$$Y(\omega) = \boldsymbol{w}(\omega)^H \boldsymbol{a}(\omega)S(\omega) = \Psi(\boldsymbol{k}, \omega)S(\omega)$$

If $(\theta_s, \phi_s) = (\theta_T, \phi_T)$, $Y(\omega) = S(\omega)$

Time delay corresponding to direction (θ_T, ϕ_T)

$$\Psi(\boldsymbol{k}, \omega) = \frac{1}{M}\sum_{m=1}^{M} \exp(j\omega\tau_m^{(T)}) \exp(-j\boldsymbol{k}^T \boldsymbol{p}_m)$$

Wavenumber-frequency response

Called a beam pattern (regarded as a function of (θ_s, ϕ_s))

Performance Analysis

- Analyze a beam pattern of a straight-shape array
  - Suppose that the steering direction is $\theta_T = 0$

Array manifold vector

$$a_m(\omega) = \exp\left(-j\left((m-1) - \frac{M-1}{2}\right)k_x d_x\right)$$

$$\boldsymbol{k} = [k_x, k_y, k_z]$$

$$k_x = -\frac{2\pi}{\lambda}\sin\theta_s$$

Visible region

$$-\frac{2\pi}{\lambda} \le k_x \le \frac{2\pi}{\lambda}$$

Steering vector

$$\boldsymbol{w}^H(\omega) = \frac{1}{M}[e^{j\omega\tau_1}, e^{j\omega\tau_2}, \cdots, e^{j\omega\tau_M}] = \frac{1}{M}[1, \cdots, 1]$$

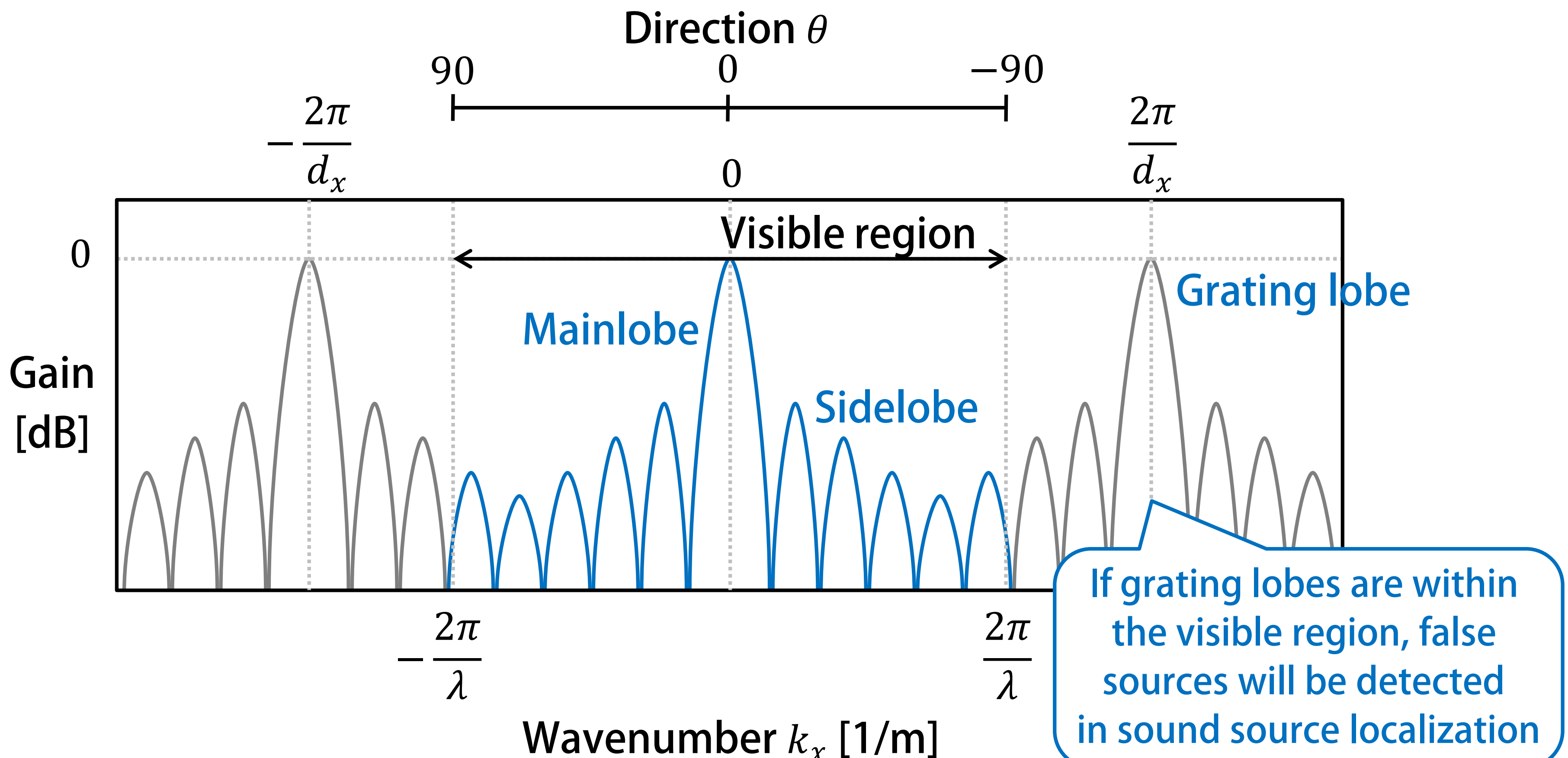
Beam pattern

$$\Psi(\boldsymbol{k}, \omega) = \boldsymbol{w}^H(\omega)\boldsymbol{a}(\omega) = \frac{1}{M}\sum_{m=1}^{M}\exp\left(-j\left((m-1) - \frac{M-1}{2}\right)k_x d_x\right) = \frac{1}{M}\frac{\sin\left(\frac{M k_x d_x}{2}\right)}{\sin\left(\frac{k_x d_x}{2}\right)}$$

Sum of geometric progression

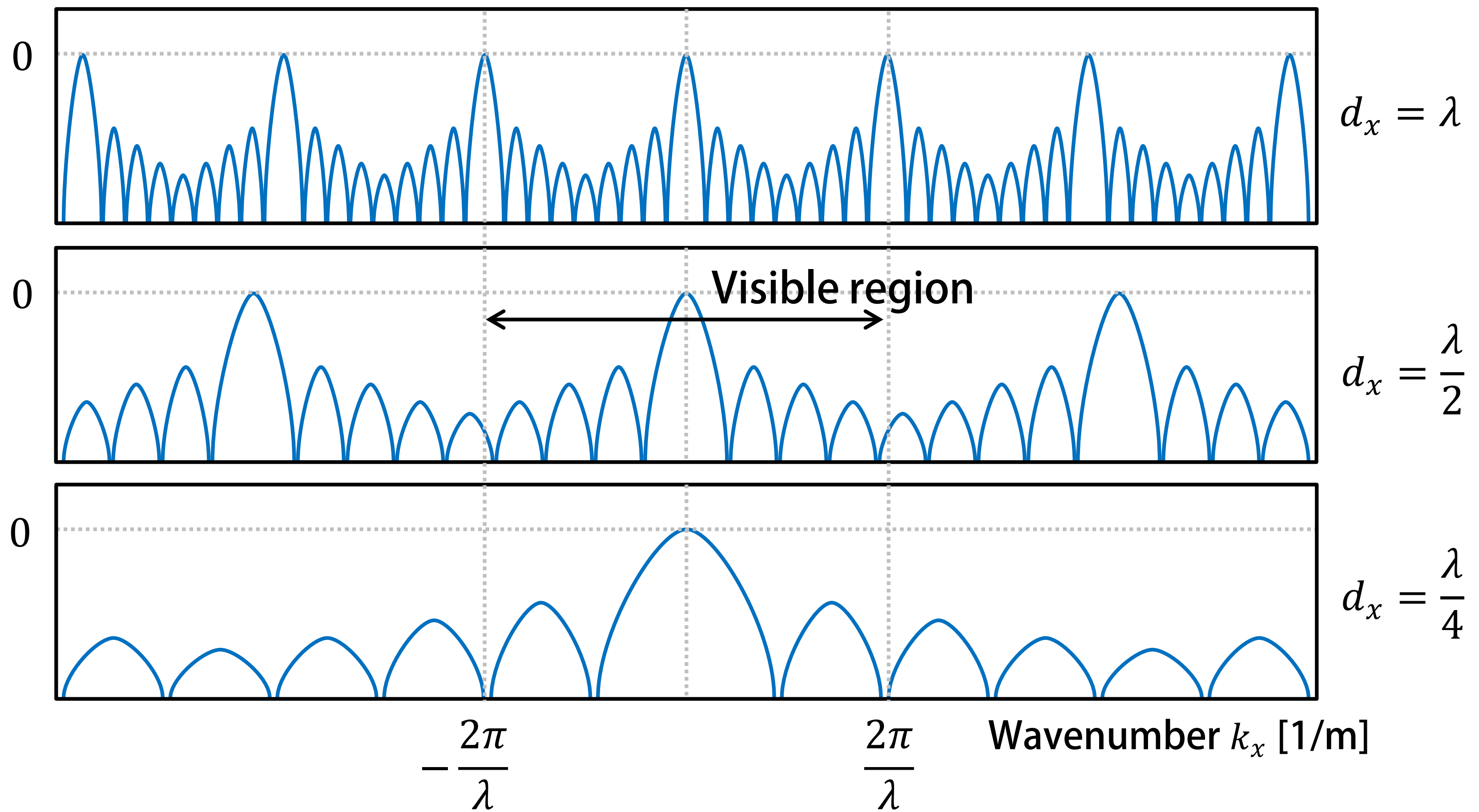
Beam Pattern

- Visualize a beam pattern: $20\log_{10}|\Psi(\boldsymbol{k},\omega)|$



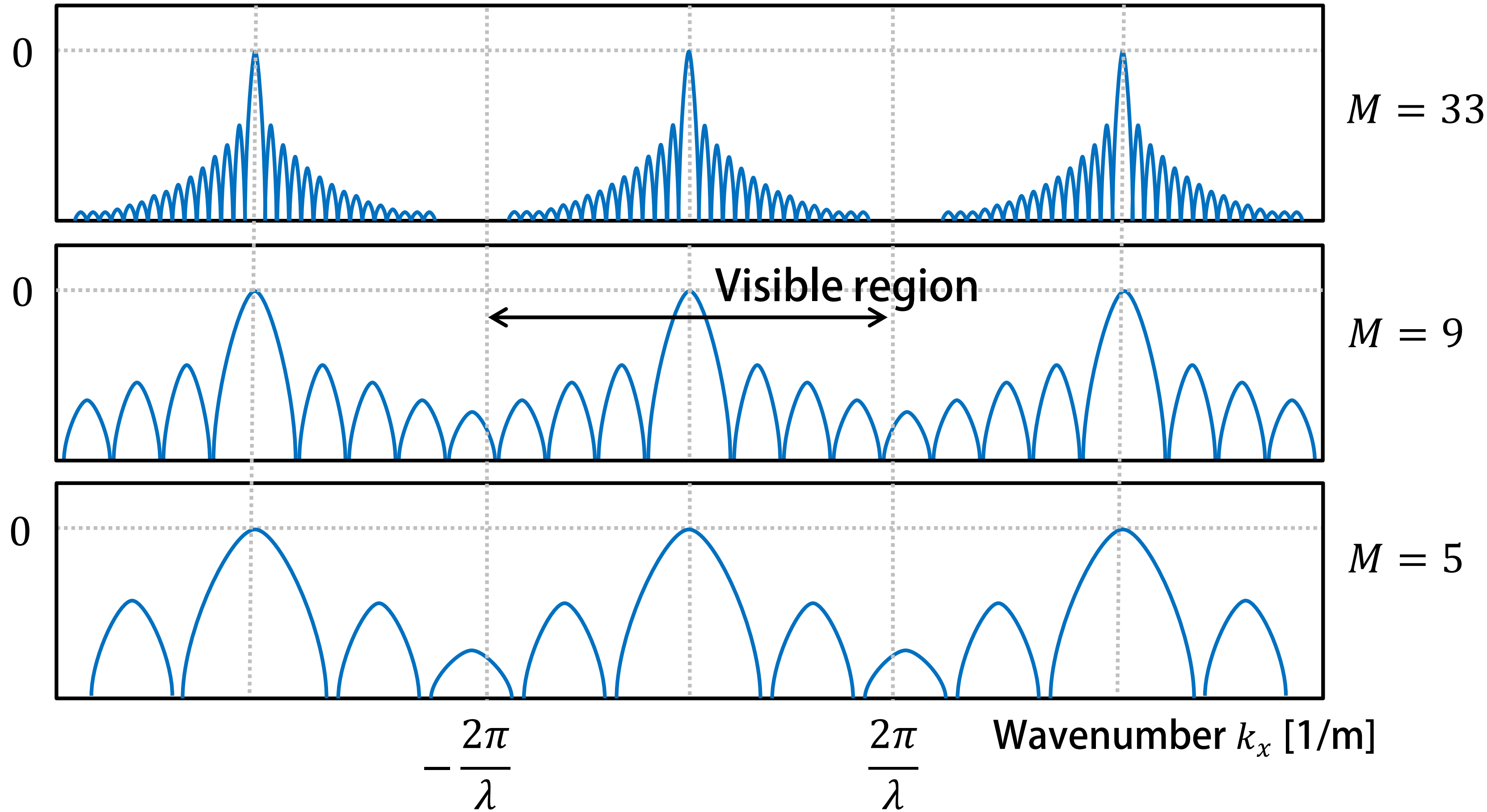
Beam Pattern

- The microphone interval d_x affects the beam pattern



Beam Pattern

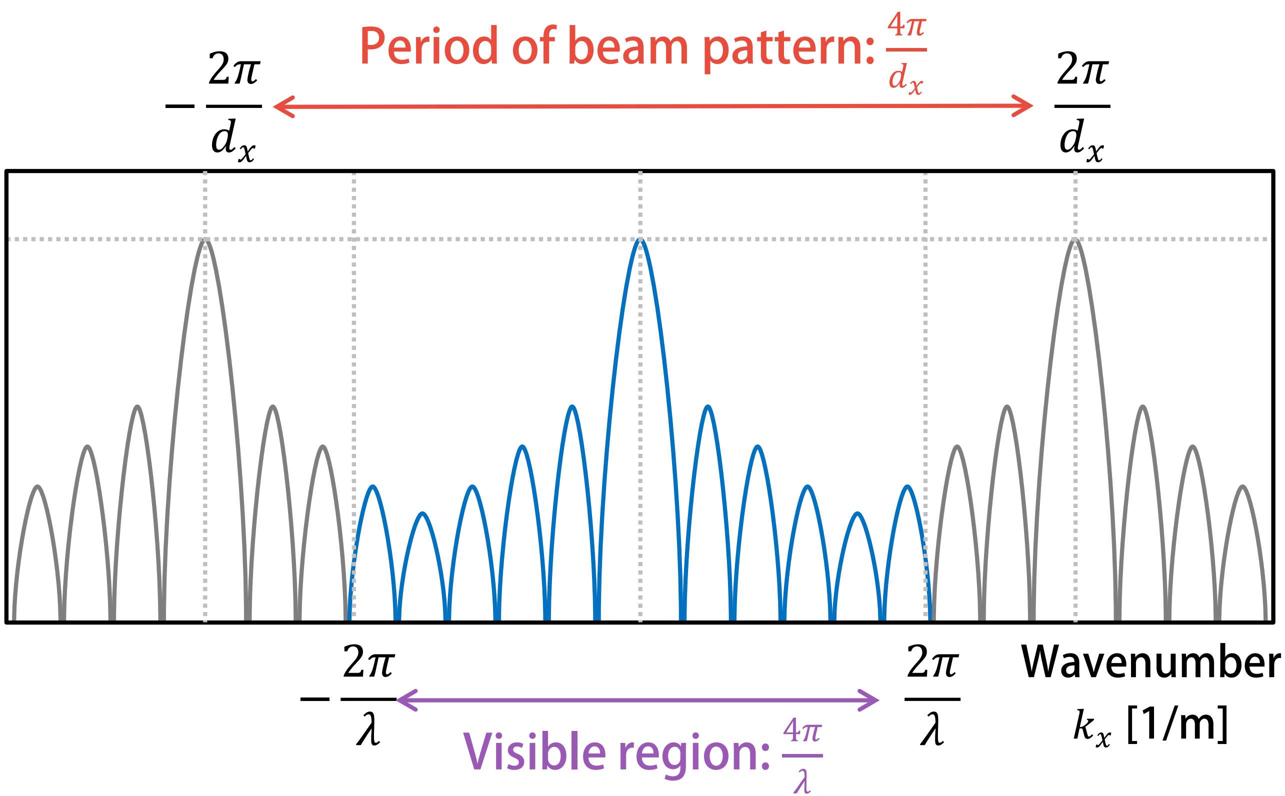
- The aperture Md_x affects the beam pattern



Geometry of Microphone Array

- The mic interval must be set for avoiding spatial aliasing
  - Grafting lobes should be without the visible region

$$\frac{2\pi}{d_x} > \frac{4\pi}{\lambda} \Rightarrow d_x \leq \frac{\lambda}{2}$$



Sampling theorem in spatial domain

$$d_x \leq \frac{c}{2f}$$

Mic interval

Sampling theorem in time domain

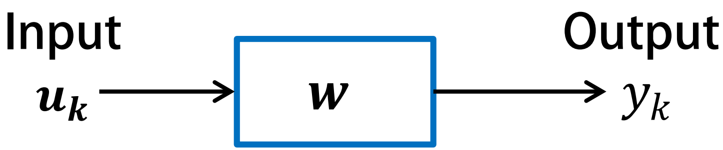
$$T_s \leq \frac{1}{2f}$$

Sampling interval

Spatial Wiener Filter

- Multichannel Wiener filter in the spatial domain

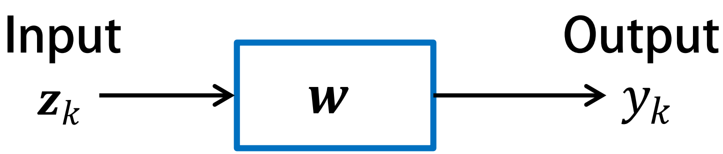
Time-domain Wiener filter



$$y_k = \boldsymbol{w}^H \boldsymbol{u}_k \qquad \boldsymbol{w} = \begin{bmatrix} w_1 \\ w_2 \\ \vdots \\ w_K \end{bmatrix} \quad \boldsymbol{u}_k = \begin{bmatrix} u_k \\ u_{k-1} \\ \vdots \\ u_{k-K+1} \end{bmatrix}$$

$$\boldsymbol{w}_{\mathrm{MF}} = \boldsymbol{R}_u^{-1} \boldsymbol{r}_{ud} \qquad \boldsymbol{R}_u = E\left[\boldsymbol{u}_k \boldsymbol{u}_k^H\right] \qquad \boldsymbol{r}_{ud} = E[\boldsymbol{u}_k d_k^*]$$

Spatial-domain Wiener filter



$$y_k = \boldsymbol{w}^H \boldsymbol{z}_k \qquad \boldsymbol{w} = \begin{bmatrix} w_1 \\ w_2 \\ \vdots \\ w_K \end{bmatrix} \quad z_k = \begin{bmatrix} Z_1(\omega, k) \\ Z_2(\omega, K) \\ \vdots \\ Z_M(\omega, k) \end{bmatrix}$$

$$\boldsymbol{w}_{\mathrm{MF}} = \boldsymbol{R}_z^{-1} \boldsymbol{r}_{zd} \qquad \boldsymbol{R}_Z = E[\boldsymbol{z}\boldsymbol{z}^H] \qquad \boldsymbol{r}_{zd} = E[\boldsymbol{z} d^*]$$

Maximum-Likelihood Beamformer

- Maximize the likelihood for observed data $\boldsymbol{z}$

Source signal → Observed signals

$$\boldsymbol{z}(\omega) = \boldsymbol{a}(\omega)S(\omega) + \boldsymbol{v}(\omega) \longrightarrow \boldsymbol{z} = \boldsymbol{a}s + \boldsymbol{v}$$

Observed signals → Source signal

$$\boldsymbol{y}(\omega) = \boldsymbol{w}^H(\omega)z(\omega) \longrightarrow y = \boldsymbol{w}^H\boldsymbol{z}$$

Limitation: we need to estimate $\boldsymbol{K}$ in advance

We assume $\boldsymbol{v} \sim N(\boldsymbol{v}|0, \boldsymbol{K})$ $\quad$ $\boldsymbol{K} = \mathrm{E}[\boldsymbol{v}\boldsymbol{v}^{\mathrm{H}}]$: correlation matrix of noise

↓ Linear transformation of $\boldsymbol{v}$

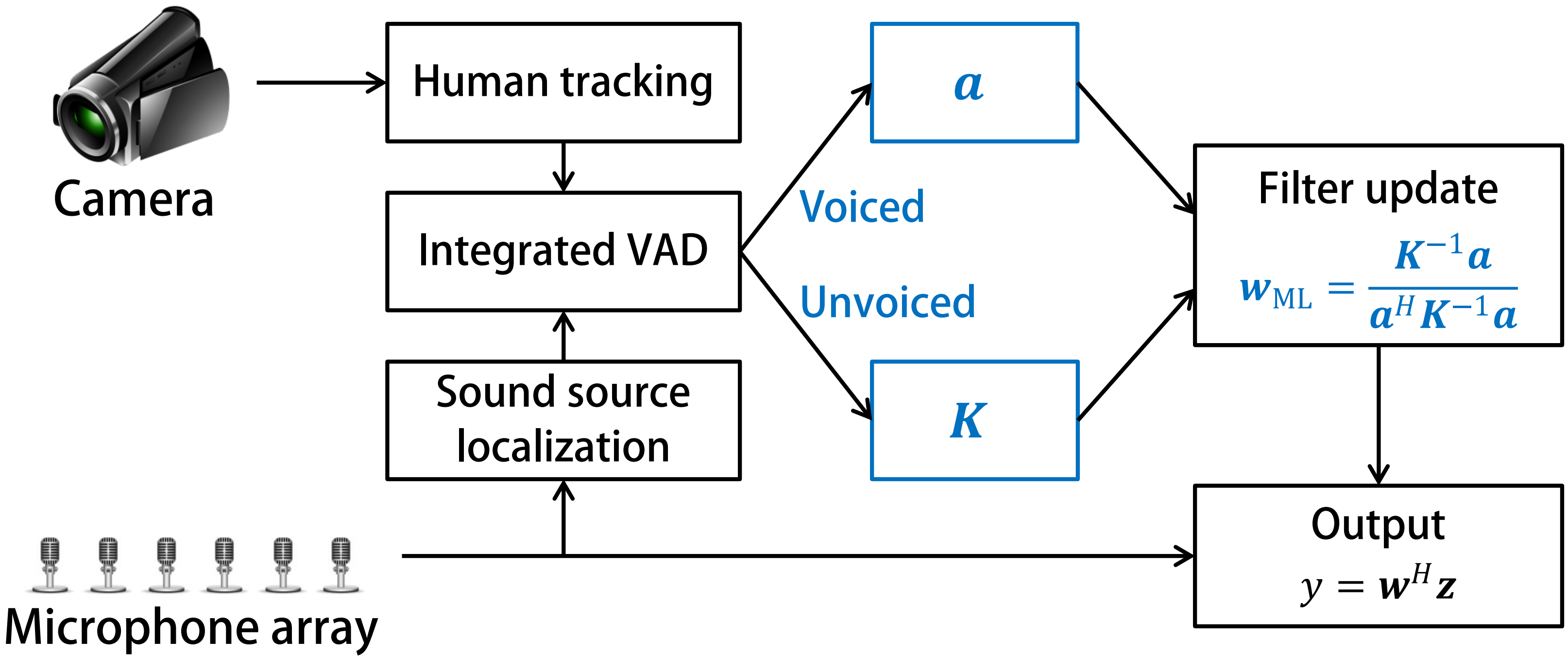
$$\boldsymbol{z}|\boldsymbol{s} \sim N(\boldsymbol{z}|\boldsymbol{a}s, \boldsymbol{K})$$

Log likelihood: $\log p(\boldsymbol{z}|\boldsymbol{s}) = -\log|\pi\boldsymbol{K}| - (\boldsymbol{z} - \boldsymbol{a}s)^H K^{-1}(\boldsymbol{z} - \boldsymbol{a}s)$

$$\frac{\partial \log p(\boldsymbol{z}|\boldsymbol{s})}{\partial \boldsymbol{s}^*} = \boldsymbol{a}^H\boldsymbol{K}^{-1}(\boldsymbol{z} - \boldsymbol{a}s) \qquad s_{\mathrm{ML}} = \frac{\boldsymbol{a}^H\boldsymbol{K}^{-1}\boldsymbol{z}}{\boldsymbol{a}^H\boldsymbol{K}^{-1}\boldsymbol{a}}(= y) \longrightarrow \boldsymbol{w}_{\mathrm{ML}} = \frac{\boldsymbol{K}^{-1}\boldsymbol{a}}{\boldsymbol{a}^H\boldsymbol{K}^{-1}\boldsymbol{a}}$$

Practical Example

- Combine ML beamformer with voice activity detection
  - Estimate an array manifold vector $\boldsymbol{a}$ for voiced regions
  - Estimate a spatial correlation matrix $\boldsymbol{K}$ for unvoiced (noise) regions



Minimum-Variance Beamformer

- Minimize the output power $|y|^2$
  - The spatial correlation matrix of noise $\boldsymbol{K}$ is not required
  - Constraint: $\boldsymbol{w}^H\boldsymbol{a} = 1$
  - Average output power: $E\left[|y|^2\right] = E\left[\left|\boldsymbol{w}^H\boldsymbol{z}\right|^2\right] = \boldsymbol{w}^H E[\boldsymbol{z}\boldsymbol{z}^H]\boldsymbol{w} = \boldsymbol{w}^H\boldsymbol{R}\boldsymbol{w}$ → Minimize

Cost function with a Lagrange multiplier λ:

$$J = \boldsymbol{w}^H\boldsymbol{R}\boldsymbol{w} + 2\mathrm{Re}\left(\lambda^*\left(\boldsymbol{a}^H\boldsymbol{w} - 1\right)\right)$$

$$= \boldsymbol{w}^H\boldsymbol{R}\boldsymbol{w} + \lambda^*\left(\boldsymbol{a}^H\boldsymbol{w} - 1\right) + \lambda\left(\boldsymbol{w}^H\boldsymbol{a} - 1\right)$$

$$\frac{\partial J}{\partial \boldsymbol{w}^*} = \boldsymbol{R}\boldsymbol{w} + \lambda\boldsymbol{a} \to 0 \qquad \boldsymbol{w}^* = -\lambda\boldsymbol{R}^{-1}\boldsymbol{a}$$

$$\lambda = -\left(\boldsymbol{a}^H\boldsymbol{R}^{-1}\boldsymbol{a}\right)^{-1} \longrightarrow \boldsymbol{w}_{\mathrm{MV}} = \frac{\boldsymbol{R}^{-1}\boldsymbol{a}}{\boldsymbol{a}^H\boldsymbol{R}^{-1}\boldsymbol{a}} \Longleftrightarrow \boldsymbol{w}_{\mathrm{ML}} = \frac{\boldsymbol{K}^{-1}\boldsymbol{a}}{\boldsymbol{a}^H\boldsymbol{K}^{-1}\boldsymbol{a}}$$

Noise correlation matrix $\boldsymbol{K}$ is replaced with observed correlation matrix $\boldsymbol{R}$

Spatial Spectrum

- We are interested in the power of a signal coming from a steering-vector direction θ_T

Beamformer: $y(\theta_T) = w^H(\theta_T)z$

Average output power:

Spatial spectrum

$$P(\theta_T) = E\left[|y(\theta_T)|^2\right] = \boldsymbol{w}^H(\theta_T)E[\boldsymbol{z}\boldsymbol{z}^H]\boldsymbol{w}(\theta_T) = \boldsymbol{w}^H(\theta_T)\boldsymbol{R}\boldsymbol{w}(\theta_T)$$

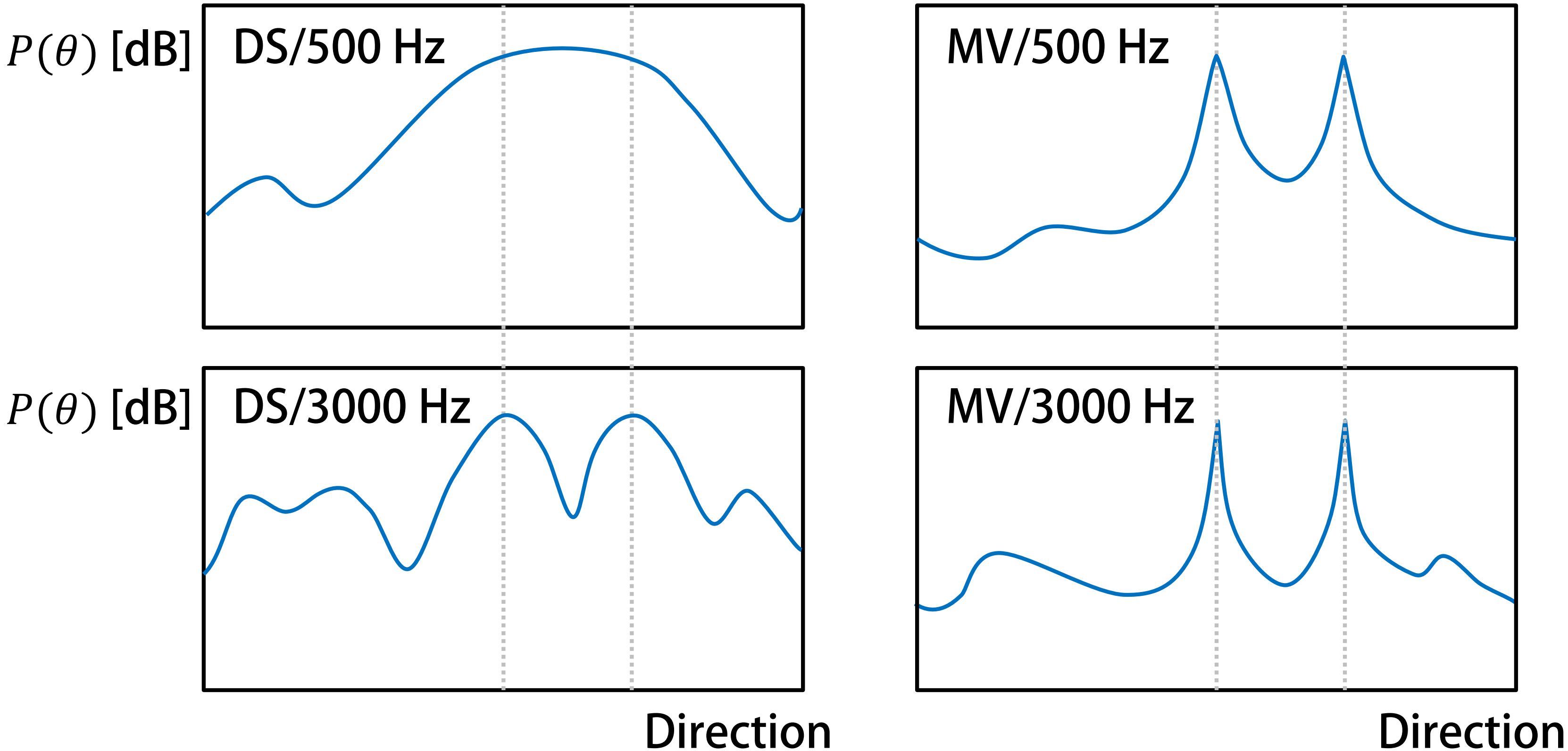
Examples:

$$\boldsymbol{w}_{\mathrm{DS}} = \frac{\boldsymbol{a}}{\boldsymbol{a}^H\boldsymbol{a}} \qquad P_{\mathrm{DS}}(\theta) = \frac{\boldsymbol{a}^H(\theta)}{\boldsymbol{a}^H(\theta)\boldsymbol{a}(\theta)}\boldsymbol{R}\frac{\boldsymbol{a}(\theta)}{\boldsymbol{a}^H(\theta)\boldsymbol{a}(\theta)} = \frac{\boldsymbol{a}^H(\theta)\boldsymbol{R}\boldsymbol{a}(\theta)}{|\boldsymbol{a}^H(\theta)\boldsymbol{a}(\theta)|^2}$$

$$\boldsymbol{w}_{\mathrm{MV}} = \frac{\boldsymbol{R}^{-1}\boldsymbol{a}}{\boldsymbol{a}^H\boldsymbol{R}^{-1}\boldsymbol{a}} \qquad P_{\mathrm{MV}}(\theta) = \frac{\boldsymbol{a}^H(\theta)\boldsymbol{R}^{-1}}{\boldsymbol{a}^H(\theta)\boldsymbol{R}^{-1}\boldsymbol{a}(\theta)}\boldsymbol{R}\frac{\boldsymbol{R}^{-1}\boldsymbol{a}(\theta)}{\boldsymbol{a}^H(\theta)\boldsymbol{R}^{-1}\boldsymbol{a}(\theta)} = \frac{1}{\boldsymbol{a}^H(\theta)\boldsymbol{R}^{-1}\boldsymbol{a}(\theta)}$$

Spatial Spectrum

- MV gives better spatial resolution than DS
  - MV has a similar property to MUSIC method (explained later)



Multiple Signal Classification (MUSIC)

Subspace Methods

- Represent an observed vector $\boldsymbol{z} \in \mathbb{C}^M$ in another space

Frequency-domain method	Eigenspace method
Fourier transform $\boldsymbol{y} = \boldsymbol{F}\boldsymbol{z}$	Karhunen-Loève transform $\boldsymbol{y} = \boldsymbol{E}^H \mathbf{z}$
Invserse Fourier transform $\boldsymbol{z} = \boldsymbol{F}^H \boldsymbol{y}$	Karhunen-Loève expansion $\boldsymbol{z} = \boldsymbol{E}\mathbf{y}$
$\boldsymbol{F}$ is a discrete transform matrix	$\boldsymbol{E} = [\boldsymbol{e}_1, \boldsymbol{e}_2, \cdots, \boldsymbol{e}_M]$ is a set of eigenvectors of $\boldsymbol{R} = E[\boldsymbol{z}\boldsymbol{z}^H]$

PCA (Karhunen-Loève transform, Karhunen-Loève expansion)

Eigenvalue decomposition
$$\boldsymbol{R}\boldsymbol{e}_i = \lambda_i \boldsymbol{e}_i$$

Spectral decomposition
$$\boldsymbol{R} = \boldsymbol{E}\boldsymbol{\Lambda}\boldsymbol{E}^H$$

$\boldsymbol{\Lambda} = \mathrm{diag}(\lambda_1, \cdots, \lambda_M)$ is a set of the corresponding eigenvalues

The average power of the i^{th} principal component
$$E\left[|y_i|^2\right] = E\left[\boldsymbol{e}_i^H \boldsymbol{z}\boldsymbol{z}^H \boldsymbol{e}_i\right] = \boldsymbol{e}_i^H \boldsymbol{R}\boldsymbol{e}_i = \lambda_i$$

Case 1: No Noise

- Observed signal = Sum of direct signals
  - Suppose that $\boldsymbol{v} = \mathbf{0}$ and $M > N$ (#microphones > #sources)

Observation model: $\boldsymbol{z} = \boldsymbol{A}\boldsymbol{s} + \boldsymbol{v}$ $\quad \boldsymbol{\Gamma} = E[\boldsymbol{s}\boldsymbol{s}^H]$

$$\boldsymbol{z} = \sum_{i=1}^{N} \boldsymbol{a}_i s_i$$

$$\boldsymbol{R} = E[\boldsymbol{z}\boldsymbol{z}^H] = E[\boldsymbol{A}\boldsymbol{s}\boldsymbol{s}^H\boldsymbol{A}^H] = \boldsymbol{A}\boldsymbol{\Gamma}\boldsymbol{A}^H$$

$$\mathrm{rank}(\boldsymbol{A}) = \mathrm{rank}(\boldsymbol{\Gamma}) = N \longrightarrow \mathrm{rank}(\boldsymbol{R}) = N$$

Eigenvalue decomposition: $\boldsymbol{R} = \boldsymbol{E}\mathbf{M}\boldsymbol{E}^H$

Eigenvalues: $\mathbf{M} = \mathrm{diag}(\mu_1, \cdots, \mu_M)$ $\quad \mu_1 > \cdots > \mu_N > 0,\ \mu_{N+1} = \cdots = \mu_M = 0$

Eigenvectors: $\boldsymbol{E} = \{\boldsymbol{e}_1, \cdots, \boldsymbol{e}_M\}$ $\quad \boldsymbol{e}_i^H\boldsymbol{R}\boldsymbol{e}_i = \mu_i$

$$\boldsymbol{e}_i^H\boldsymbol{R}\boldsymbol{e}_i = \boldsymbol{e}_i^H\boldsymbol{A}\boldsymbol{\Gamma}\boldsymbol{A}^H\boldsymbol{e}_i = \left(\boldsymbol{A}^H\boldsymbol{e}_i\right)^H\boldsymbol{\Gamma}\left(\boldsymbol{A}^H\boldsymbol{e}_i\right) = \mu_i$$

Orthogonal relationships

$$\boldsymbol{A}^H\boldsymbol{e}_i = \mathbf{0}_{N\times 1}\ (N < i \le M) \longrightarrow \boldsymbol{a}_j^H\boldsymbol{e}_i = 0\ (1 \le j \le N, N < i \le M)$$

Signal and Noise Subspaces

- Orthogonal-complementary subspaces of $\boldsymbol{A}$
  - Column space: $\mathcal{R}(\boldsymbol{A}) = \mathrm{span}(\boldsymbol{a}_1, \cdots, \boldsymbol{a}_N) \rightarrow$ Signal subspace
  - Left nullspace: $N\left(\boldsymbol{A}^H\right) = \mathrm{span}(\boldsymbol{e}_{N+1}, \cdots, \boldsymbol{e}_M) \rightarrow$ Noise subspace

$$\boldsymbol{z} = \boldsymbol{A}\boldsymbol{s}$$

μ_i: the power of signal $\boldsymbol{s}$ in the i^{th} subspace

Eigenvalue decomposition

$$\boldsymbol{R} = E[\boldsymbol{z}\boldsymbol{z}^H] = [\boldsymbol{e}_1, \boldsymbol{e}_2, \cdots, \boldsymbol{e}_M]\mathrm{diag}(\mu_1, \mu_2, \cdots, \mu_M)[\boldsymbol{e}_1, \boldsymbol{e}_2, \cdots, \boldsymbol{e}_M]^H$$

$$\mathrm{span}(\boldsymbol{e}_1, \cdots, \boldsymbol{e}_N) = \mathrm{span}(\boldsymbol{e}_{N+1}, \cdots, \boldsymbol{e}_M)^{\perp}$$

Orthogonal bases

Identical

Result of the previous slide

$$\boldsymbol{a}_j^H \boldsymbol{e}_i = 0 \ (1 \le j \le N, N < i \le M)$$

$$\mathrm{span}(\boldsymbol{a}_1, \cdots, \boldsymbol{a}_N) = \mathrm{span}(\boldsymbol{e}_{N+1}, \cdots, \boldsymbol{e}_M)^{\perp}$$

Case 2: White Noise

- Observed signal = Sum of direct signals + White noise
  - Suppose that $\boldsymbol{v} = \boldsymbol{v}_w$ and $M > N$

Observation model: $\boldsymbol{z} = \boldsymbol{A}\boldsymbol{s} + \boldsymbol{v}_w$ $\quad \boldsymbol{\Gamma} = E[\boldsymbol{s}\boldsymbol{s}^H]$ $\quad \sigma^2 \boldsymbol{I} = E[\boldsymbol{v}_w \boldsymbol{v}_w^H]$

$$\boldsymbol{z} = \sum_{i=1}^{N} \boldsymbol{a}_i s_i + \boldsymbol{v}_w$$

$$\boldsymbol{R} = E[\boldsymbol{z}\boldsymbol{z}^H] = \boldsymbol{A}\boldsymbol{\Gamma}\boldsymbol{A}^H + \sigma^2 \boldsymbol{I}$$

$$\mathrm{rank}(\boldsymbol{A}) = \mathrm{rank}(\boldsymbol{\Gamma}) = N \longrightarrow \mathrm{rank}(\boldsymbol{R}) = N$$

Eigenvalue decomposition: $\boldsymbol{R} = \boldsymbol{E}\boldsymbol{\Lambda}\boldsymbol{E}^H$

Eigenvalues: $\boldsymbol{\Lambda} = \mathrm{diag}(\lambda_1, \cdots, \lambda_M)$ $\qquad \boldsymbol{\Lambda} = \mathbf{M} + \sigma^2 \boldsymbol{I}$ ← No-noise case + $\sigma^2 \boldsymbol{I}$

Eigenvectors: $\boldsymbol{E} = \{\boldsymbol{e}_1, \cdots, \boldsymbol{e}_M\}$ $\qquad \boldsymbol{e}_i^H \boldsymbol{R} \boldsymbol{e}_i = \lambda_i$

$$\boldsymbol{e}_i^H \boldsymbol{R} \boldsymbol{e}_i = \boldsymbol{e}_i^H (\boldsymbol{A}\boldsymbol{\Gamma}\boldsymbol{A}^H + \sigma^2 \boldsymbol{I}) \boldsymbol{e}_i = (\boldsymbol{A}^H \boldsymbol{e}_i)^H \boldsymbol{\Gamma} (\boldsymbol{A}^H \boldsymbol{e}_i) + \sigma^2$$

$$\boldsymbol{A}^H \boldsymbol{e}_i = \boldsymbol{0}_{N\times 1} \ (N < i \le M) \longrightarrow \boldsymbol{a}_j^H \boldsymbol{e}_i = 0 \ (1 \le j \le N, N < i \le M)$$

Orthogonal relationships

Signal and Noise Subspaces

- Orthogonal-complementary subspaces of $\boldsymbol{A}$
  - Column space: $\mathcal{R}(\boldsymbol{A}) = \text{span}(\boldsymbol{a}_1, \cdots, \boldsymbol{a}_N) \rightarrow$ Signal subspace
  - Left nullspace: $N\left(\boldsymbol{A}^H\right) = \text{span}(\boldsymbol{e}_{N+1}, \cdots, \boldsymbol{e}_M) \rightarrow$ Noise subspace

$$\boldsymbol{z} = \boldsymbol{A}\boldsymbol{s} + \boldsymbol{v}_w$$

λ_i: the sum of the power of signal s and noise $\boldsymbol{v}_w$ in the i^{th} subspace

Eigenvalue decomposition

$$\boldsymbol{R} = E[\boldsymbol{z}\boldsymbol{z}^H] = [\boldsymbol{e}_1, \boldsymbol{e}_2, \cdots, \boldsymbol{e}_M]\text{diag}(\lambda_1, \lambda_2, \cdots, \lambda_M)[\boldsymbol{e}_1, \boldsymbol{e}_2, \cdots, \boldsymbol{e}_M]^H$$

$$\text{span}(\boldsymbol{e}_1, \cdots, \boldsymbol{e}_N) = \text{span}(\boldsymbol{e}_{N+1}, \cdots, \boldsymbol{e}_M)^{\perp}$$

Orthogonal bases

Identical

Result of the previous slide

$$\boldsymbol{a}_j^H \boldsymbol{e}_i = 0 \; (1 \leq j \leq N, N < i \leq M)$$

$$\text{span}(\boldsymbol{a}_1, \cdots, \boldsymbol{a}_N) = \text{span}(\boldsymbol{e}_{N+1}, \cdots, \boldsymbol{e}_M)^{\perp}$$

Case 3: Colored Noise

- Observed signal = Sum of direct signals + Colored noise
  - Suppose that $\boldsymbol{v} = \boldsymbol{v}_c$ and $M > N$

Non-diagonal matrix

Observation model: $\boldsymbol{z} = \boldsymbol{A}\boldsymbol{s} + \boldsymbol{v}_c$ $\quad \boldsymbol{\Gamma} = E[\boldsymbol{s}\boldsymbol{s}^H]$ $\quad \boldsymbol{K} = E[\boldsymbol{v}_c\boldsymbol{v}_c^H]$

$$\boldsymbol{z} = \sum_{i=1}^{N} \boldsymbol{a}_i s_i + \boldsymbol{v}_c$$

$$\boldsymbol{R} = E[\boldsymbol{z}\boldsymbol{z}^H] = \boldsymbol{A}\boldsymbol{\Gamma}\boldsymbol{A}^H + \boldsymbol{K}$$

$$\mathrm{rank}(\boldsymbol{A}) = \mathrm{rank}(\boldsymbol{\Gamma}) = N \longrightarrow \mathrm{rank}(\boldsymbol{R}) = N$$

Generalized eigenvalue decomp. of $\boldsymbol{R}$

$$\boldsymbol{R}\boldsymbol{e}_i = \lambda_i \boldsymbol{K}\boldsymbol{e}_i$$

Eigenvalues: $\boldsymbol{\Lambda} = \{\lambda_1, \cdots, \lambda_M\}$

Eigenvectors: $\boldsymbol{E} = \{\boldsymbol{e}_1, \cdots, \boldsymbol{e}_M\}$

⇔

Eigenvalue decomp. of $\boldsymbol{\Phi}^{-\mathrm{H}}\boldsymbol{R}\boldsymbol{\Phi}^{-1}$

$$\left(\boldsymbol{\Phi}^{-H}\boldsymbol{R}\boldsymbol{\Phi}^{-1}\right)\boldsymbol{f}_i = \lambda_i \boldsymbol{f}_i$$

Eigenvalues: $\boldsymbol{\Lambda} = \{\lambda_1, \cdots, \lambda_M\}$

Eigenvectors: $\boldsymbol{F} = \{\boldsymbol{f}_1, \cdots, \boldsymbol{f}_M\}$

$\boldsymbol{\Phi}^H\boldsymbol{\Phi} = \boldsymbol{K}$ $\quad \boldsymbol{f}_i = \boldsymbol{\Phi}\boldsymbol{e}_i$ $\quad \mathrm{rank}\left(\boldsymbol{\Phi}^{-H}\boldsymbol{R}\boldsymbol{\Phi}^{-1}\right) = N$

Signal and Noise Subspaces

- Orthogonal-complementary subspaces of $\boldsymbol{A}$ $\boldsymbol{\Gamma} = E[\boldsymbol{s}\boldsymbol{s}^H]$

	No-noise case $\boldsymbol{z} = \boldsymbol{A}\boldsymbol{s}$		White noise $\boldsymbol{z} = \boldsymbol{A}\boldsymbol{s} + \boldsymbol{v}_w$		Colored noise $\boldsymbol{z} = \boldsymbol{A}\boldsymbol{s} + \boldsymbol{v}_c$	
	Signal power	Noise power	Signal power	Noise power	Signal power	Noise power
Signal subspace $(1 \le i \le N)$	μ_i	0	μ_i	σ^2	$\check{\mu}_i$	1
Noise subspace $(N < i \le M)$	0	0	0	σ^2	0	1
$E[\boldsymbol{z}\boldsymbol{z}^H](= \boldsymbol{R})$	$\boldsymbol{A}\boldsymbol{\Gamma}\boldsymbol{A}^H$		$\boldsymbol{A}\boldsymbol{\Gamma}\boldsymbol{A}^H + \sigma^2\boldsymbol{I}$		$\boldsymbol{A}\boldsymbol{\Gamma}\boldsymbol{A}^H + \sigma^2\boldsymbol{K}$	
$E[\boldsymbol{v}\boldsymbol{v}^H]$	$\boldsymbol{0}$		$\sigma^2\boldsymbol{I}$		$\boldsymbol{K} = \boldsymbol{\Phi}^H\boldsymbol{\Phi}$	
Eigenvalue decomposition	$\boldsymbol{R} = \boldsymbol{E}\mathbf{M}\boldsymbol{E}^H$		$\boldsymbol{R} = \boldsymbol{E}\boldsymbol{\Lambda}\boldsymbol{E}^H$		$\boldsymbol{\Phi}^{-H}\boldsymbol{R}\boldsymbol{\Phi}^{-1} = \boldsymbol{F}\check{\boldsymbol{\Lambda}}\boldsymbol{F}^H$	

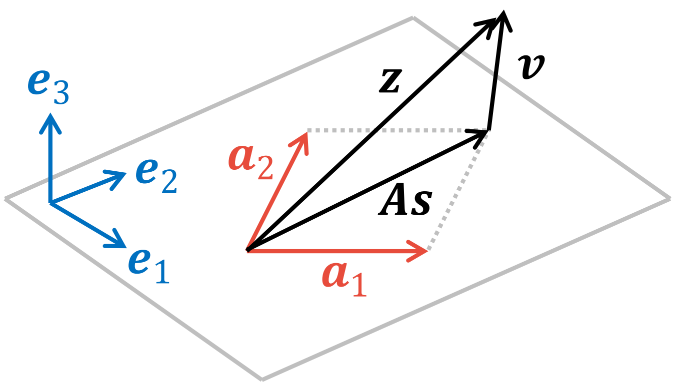
MUSIC

- Adaptive beamforming based on subspace analysis
  - Separate signal and nose components into different subspaces
  - Calculate spatial spectrum $P_{\mathrm{MUSIC}}(\theta)$

$$P_{\mathrm{MUS}}(\theta) = \frac{\|\boldsymbol{a}(\theta)\|^2}{\sum_{i=N+1}^{M} |\boldsymbol{a}^H(\theta)\boldsymbol{e}_i|^2} = \frac{\boldsymbol{a}^H(\theta)\boldsymbol{a}(\theta)}{\boldsymbol{a}^H(\theta)\boldsymbol{E}_n\boldsymbol{E}_n^H\boldsymbol{a}(\theta)}$$

$\boldsymbol{E}_n = [\boldsymbol{e}_{N+1}, \cdots, \boldsymbol{e}_M]$: a set of eigenvectors corresponding noise subspaces

$\boldsymbol{a}(\theta)$: array manifold vector (θ: *assumed* source direction)



If θ matches a true source direction $(\boldsymbol{a}(\theta) = \boldsymbol{a}_i)$,

$\boldsymbol{a}^H(\theta)\boldsymbol{E}_n = \boldsymbol{0}$ *i.e.*, $P_{\mathrm{MUS}}(\theta) = \infty$

Signal and noise subspaces are orthogonal

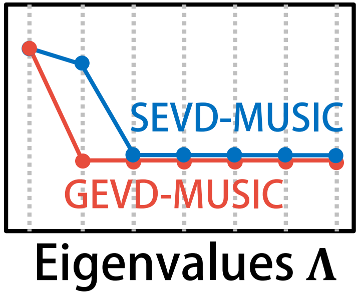
Methods

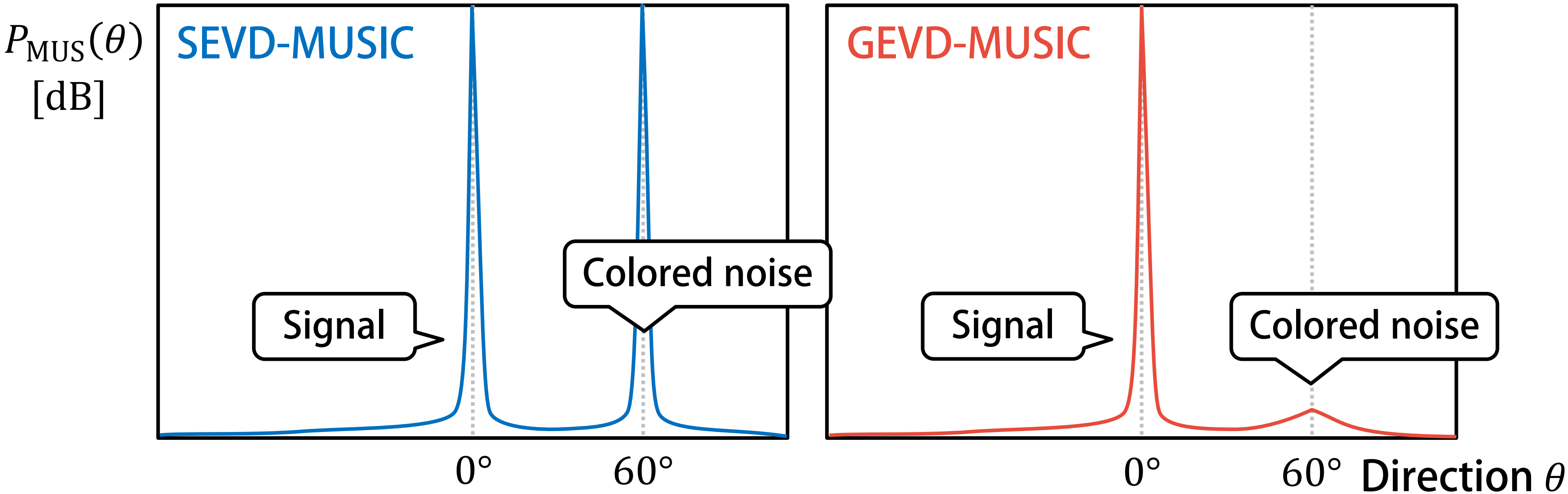
- Orthogonal-complementary subspaces of $\boldsymbol{A}$ $\quad \boldsymbol{\Gamma} = E[\boldsymbol{s}\boldsymbol{s}^H]$

	SEVD-MUSIC $\boldsymbol{z} = \boldsymbol{A}\boldsymbol{s} + \boldsymbol{v}_w$		GEVD-MUSIC $\boldsymbol{z} = \boldsymbol{A}\boldsymbol{s} + \boldsymbol{v}_c$		GSVD-MUSIC $\boldsymbol{z} = \boldsymbol{A}\boldsymbol{s} + \boldsymbol{v}_c$	
	Signal power	Noise power	Signal power	Noise power	Signal power	Noise power
Signal subspace $(1 \leq i \leq N)$	μ_i	σ^2	$\check{\mu}_i$	1	$\check{\mu}_i$	1
Noise subspace $(N < i \leq M)$	0	σ^2	0	1	0	1
$E[\boldsymbol{z}\boldsymbol{z}^H](= \boldsymbol{R})$	$\boldsymbol{A}\boldsymbol{\Gamma}\boldsymbol{A}^H + \sigma^2\boldsymbol{I}$		$\boldsymbol{A}\boldsymbol{\Gamma}\boldsymbol{A}^H + \sigma^2\boldsymbol{K}$		$\boldsymbol{A}\boldsymbol{\Gamma}\boldsymbol{A}^H + \sigma^2\boldsymbol{K}$	
$E[\boldsymbol{v}\boldsymbol{v}^H]$	$\sigma^2\boldsymbol{I}$		$\boldsymbol{K} = \boldsymbol{\Phi}^H\boldsymbol{\Phi}$		$\boldsymbol{K} = \boldsymbol{U}^H\boldsymbol{V}$	
Eigenvalue decomposition	$\boldsymbol{R} = \boldsymbol{E}\boldsymbol{\Lambda}\boldsymbol{E}^H$		$\boldsymbol{\Phi}^{-H}\boldsymbol{R}\boldsymbol{\Phi}^{-1} = \boldsymbol{F}\boldsymbol{\Lambda}\boldsymbol{F}^H$		$\boldsymbol{K}^{-1}\boldsymbol{R} = \boldsymbol{U}\boldsymbol{\Lambda}\boldsymbol{V}^{-H}$	

Spatial Spectrum

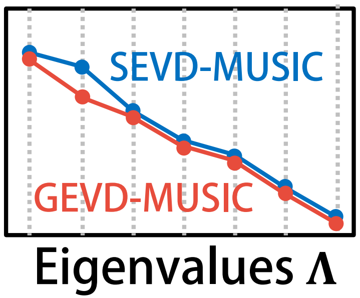
- Compare MUSIC methods in a simulated environment
  - Assume an observation model: $\boldsymbol{z} = \boldsymbol{z}_s + \boldsymbol{v}_c + \boldsymbol{v}_w$
    - Direct signal: $\boldsymbol{z}_s = \boldsymbol{a}_1 s_1$ (direction $0°$)
    - Colored noise: $\boldsymbol{v}_c = \boldsymbol{a}_1^c s_1^c$ (direction $60°$)

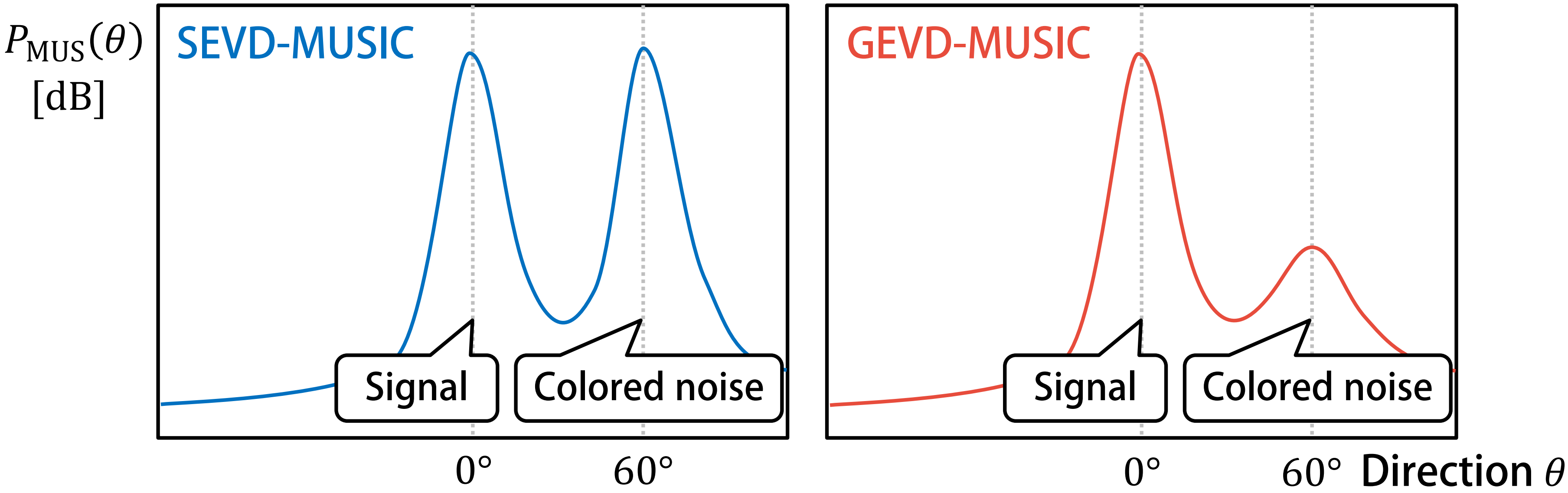




Spatial Spectrum

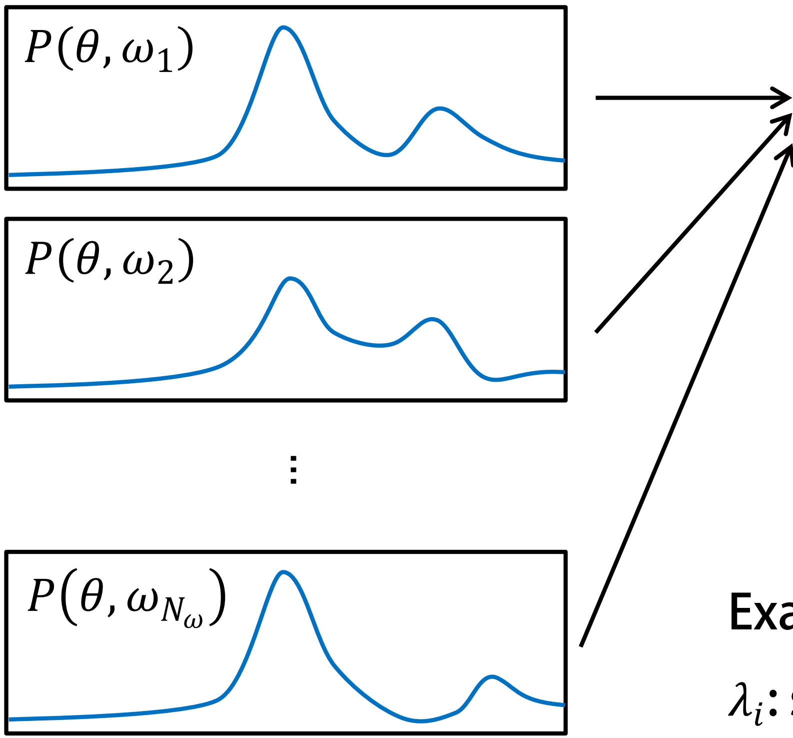
- Compare MUSIC methods in a real environment
  - Assume an observation model: $\boldsymbol{z} = \boldsymbol{z}_s + \boldsymbol{v}_c + \boldsymbol{v}_w$
    - Direct signal: $\boldsymbol{z}_s = \boldsymbol{a}_1 s_1$ (direction $0°$)
    - Colored noise: $\boldsymbol{v}_c = \boldsymbol{a}_1^c s_1^c$ (direction $60°$)

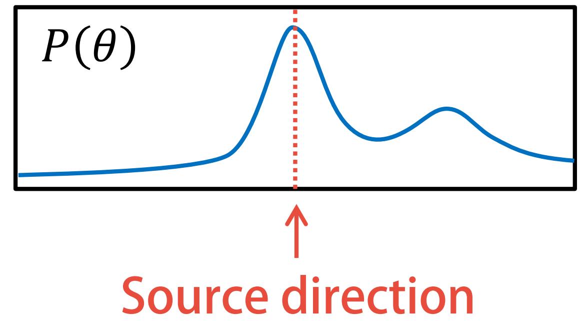




Integration Over All Frequencies

- Take the average of spatial spectra over all frequencies
  - Frequency weights $\boldsymbol{\beta}$ are determined according to an application





$$P(\theta) = \frac{1}{N_\omega} \sum_{l=1}^{N_w} \beta_l P(\theta, \omega_l)$$

Example: $\beta_l = \left[\sum_{i=1}^{N} \lambda_i(\omega_i)\right]^\alpha$

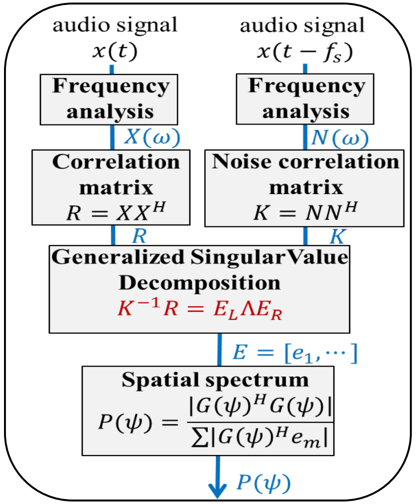
λ_i: sum of eigenvalues of signal subspace

Comparison of SSL

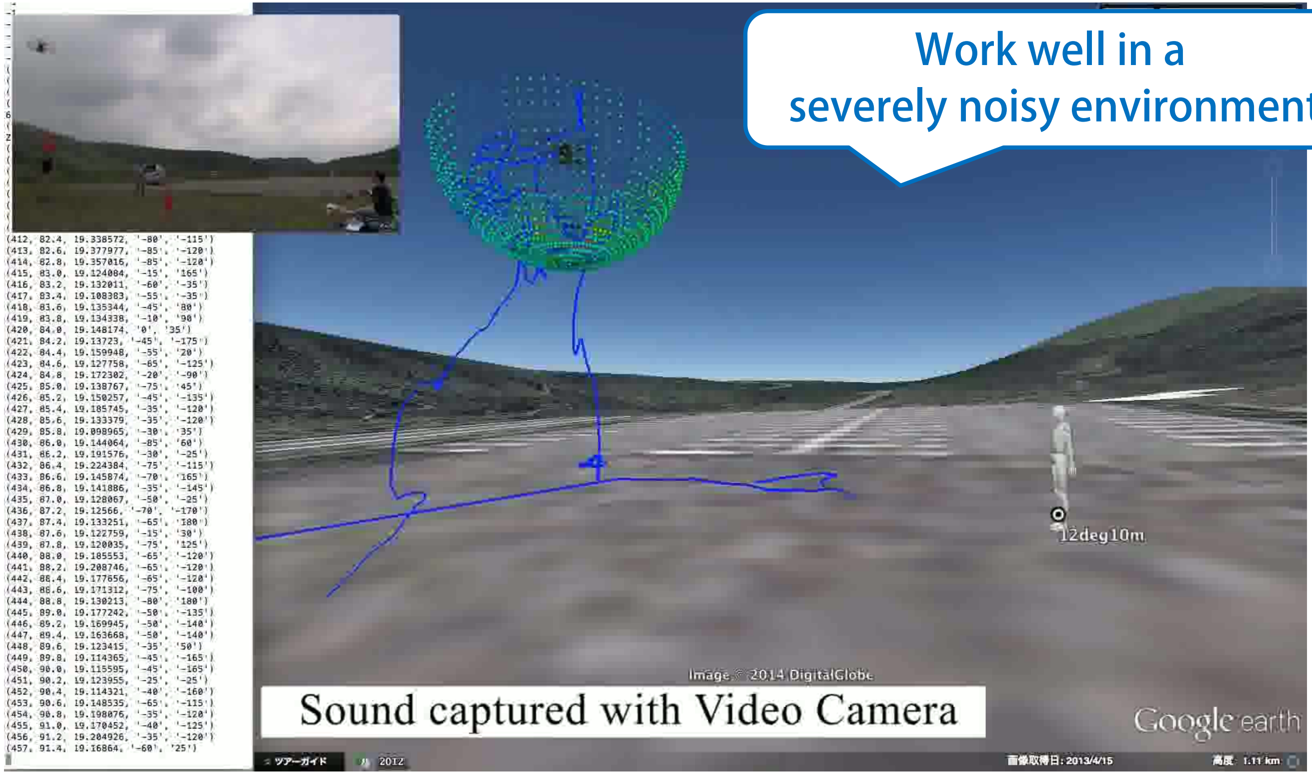
SEVD-MUSIC
and
GSVD-MUSIC

MUSIC with Adaptive Noise Estimation

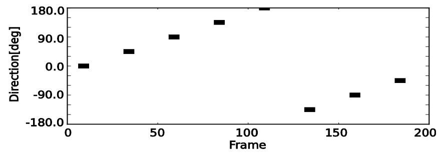
Quadrocopter with 16 mics



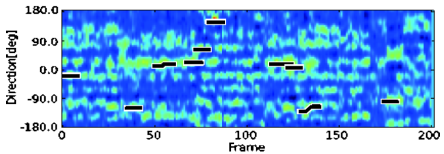
SSL with iGSVD-MUSIC



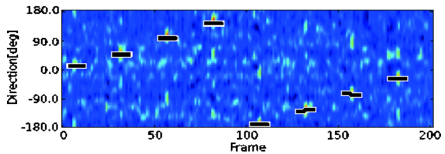
Work well in a severely noisy environment



Ground truth



SEVD-MUSIC



iGSVD-MUSIC
with adaptive noise estimation

Independent Component Analysis

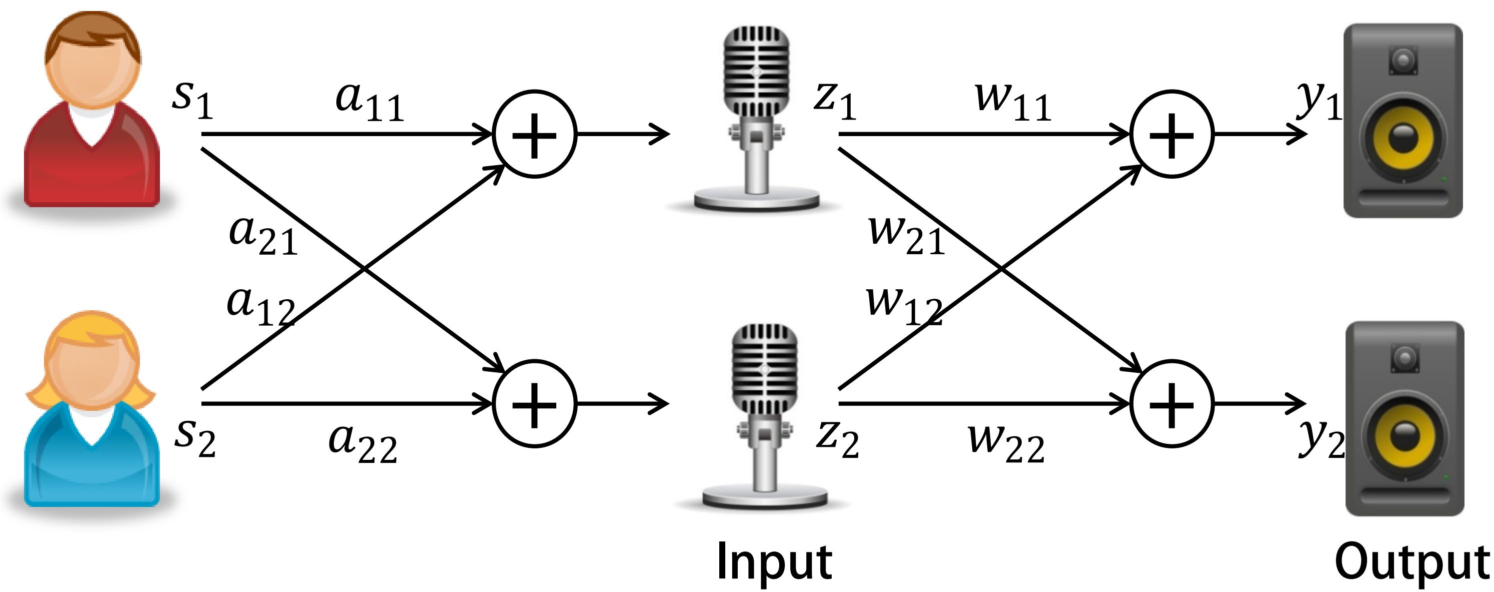
Blind Source Separation

- BSS is a mathematically ill-defined problem
  - We cannot uniquely determine source signals
    if neither prior knowledge nor constraints are taken into account
- Focus on some properties of audio signals
  - Acoustic characteristics
    - Speech: voice timbres, accent, intonation, ...
    - Musical instruments: pitches, timbres, rhythms, repetitions, ...
  - Spatial characteristics
    - Source direction (angle and elevation)

Linear methods: beamformer, independent component analysis (ICA)
Nonlinear methods: time-frequency masking

Goal

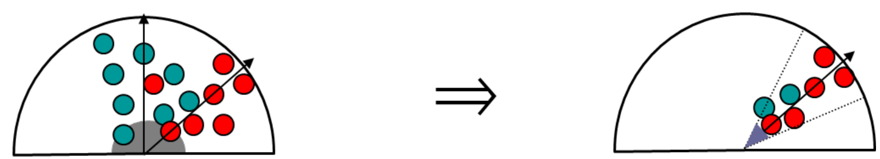
- We aim to sound source separation and localization
  - Input: $x_1, x_2, \cdots, x_N$ Output: $y_1, y_2, \cdots, y_M$ ($\approx s_1, s_2, \cdots, s_M$)
    - Mixing process: sources $s_1, s_2, \cdots, s_M \rightarrow$ observations $z_1, z_2, \cdots, z_N$
    - Two settings: A is given (non-blind) $\leftrightarrow$ A is not given (blind)



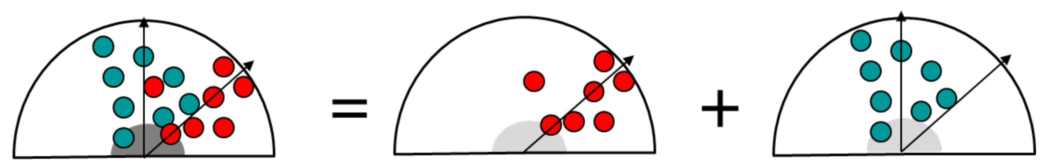
Beamforming vs. Blind Source Separation

	Beamforming	Blind source separation
Transfer functions	Required	Not necessary
Performance	Low	High
Reverberation	Can be suppressed to some extent	Included in separated signals
Issues		Permutation problem Scaling problem

Beamformer



Blind source separation



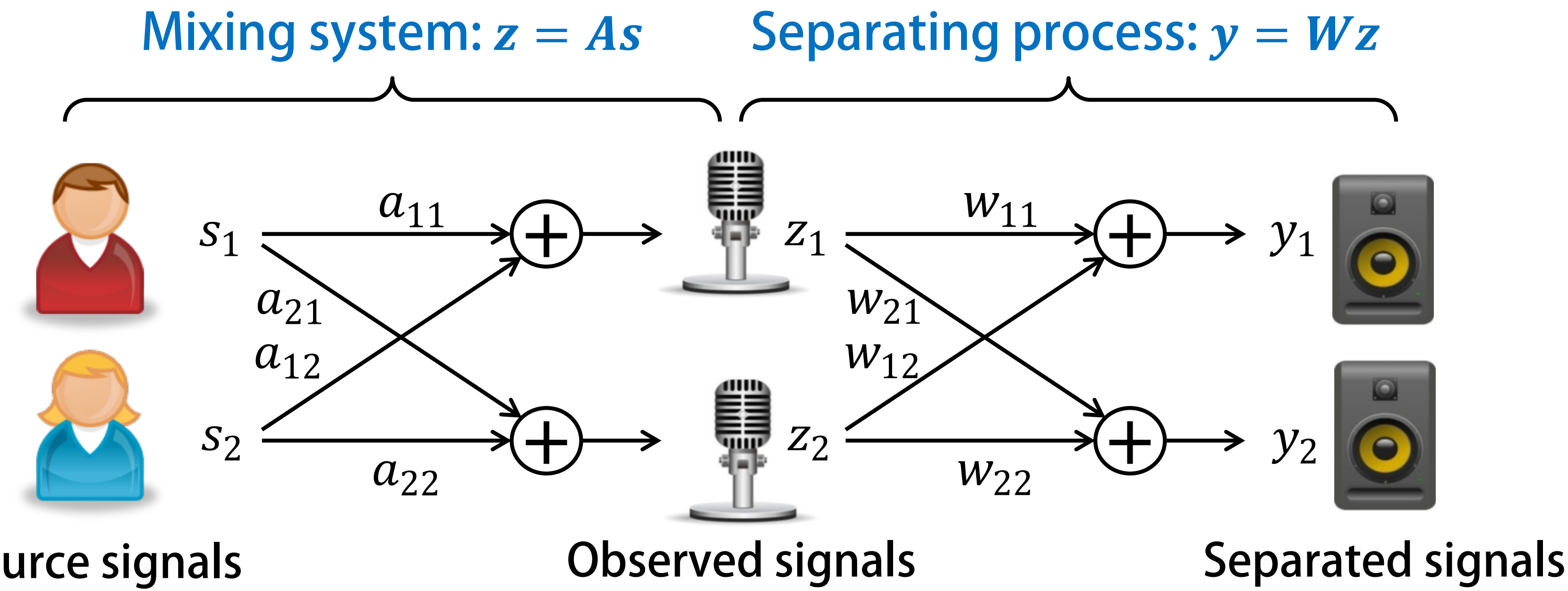
Basic Formulation

- Formulate a mixing process in the frequency domain
  - N sound sources are observed by M microphones

$M = N$ is assumed

$$\boldsymbol{z} = \boldsymbol{A}\boldsymbol{s} = \sum_{i=1}^{N} \boldsymbol{a}_i s_i \qquad \boldsymbol{y} = \boldsymbol{W}\boldsymbol{z} = \boldsymbol{W}\boldsymbol{A}\boldsymbol{s}$$

if $\boldsymbol{W} = \boldsymbol{A}^{-1}, \boldsymbol{y} \approx \boldsymbol{s}$



Principle Component Analysis

- Linearly transform an observed space into a latent space

z → $y = Wz$ → y

Observed vector — Output vector

$$W = [w_1, w_2, \cdots, w_M]^T$$

First eigenvector e_1 of R_z

First principal component

Estimate w_1 such that the variance of $y_1 = w_1^H z$ is maximized

$$E\left[|y_1|^2\right] = w_1^H E[zz^H] w_1 = w_1^H R_Z w_1 \qquad \|w_1\| = 1$$

Cost function: $J = w_1^H R_Z w_1 + \lambda_1 (1 - w_1^H w_1)$

$$\frac{\partial J}{\partial w_1^*} = R_z w_1 - \lambda_1 w_1 \to 0 \qquad E\left[|y_1|^2\right] = w_1^H R_Z w_1 = \lambda w_1^H w_1 = \lambda_1$$

λ_1 is the maximum eigenvalue & w_1 is the corresponding eigenvector

Principle Component Analysis

- The dimensions of a latent space should be orthogonal

Second eigenvector $\boldsymbol{e}_2$ of $\boldsymbol{R}_z$ — Second principal component

Estimate $\boldsymbol{w}_2$ such that the variance of $y_2 = \boldsymbol{w}_2^H \boldsymbol{z}$ is maximized

$$E\left[|y_2|^2\right] = \boldsymbol{w}_2^H E[\boldsymbol{z}\boldsymbol{z}^H]\boldsymbol{w}_2 = \boldsymbol{w}_2^H \boldsymbol{R}_Z \boldsymbol{w}_2 \qquad \|\boldsymbol{w}_2\| = 1 \ \& \ \boldsymbol{w}_1^H \boldsymbol{w}_2 = 0$$

Third eigenvector $\boldsymbol{e}_3$ of $\boldsymbol{R}_z$ — Third principal component

Estimate $\boldsymbol{w}_3$ such that the variance of $y_3 = \boldsymbol{w}_3^H \boldsymbol{z}$ is maximized

$$E\left[|y_3|^2\right] = \boldsymbol{w}_3^H E[\boldsymbol{z}\boldsymbol{z}^H]\boldsymbol{w}_3 = \boldsymbol{w}_3^H \boldsymbol{R}_Z \boldsymbol{w}_3 \qquad \|\boldsymbol{w}_3\| = 1 \ \& \ \boldsymbol{w}_1 \perp \boldsymbol{w}_2 \perp \boldsymbol{w}_3$$

Eigenvalue decomposition

$$\boldsymbol{R}_z = E[\boldsymbol{z}\boldsymbol{z}^H]$$

→

Eigenvectors: $\boldsymbol{E} = [\boldsymbol{e}_1, \cdots, \boldsymbol{e}_M]$

Eigenvalues: $\boldsymbol{\Lambda} = [\lambda_1, \cdots, \lambda_M]$

PCA: $\boldsymbol{y} = \boldsymbol{E}^H \boldsymbol{z}$ PCA with dimensionality reduction: $\boldsymbol{y} = \boldsymbol{E}_{1:p}^H \boldsymbol{z}$

Whitening

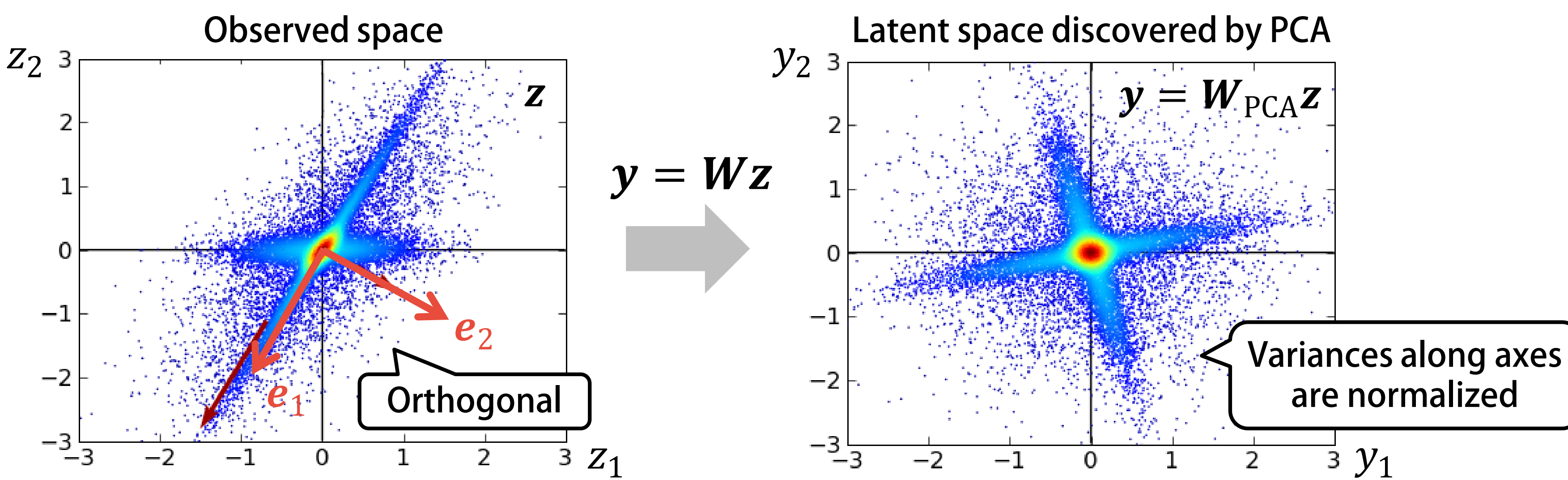
- Perform linear transform $\boldsymbol{y} = \boldsymbol{W}\boldsymbol{z}$ such that $E[\boldsymbol{y}\boldsymbol{y}^H] = 0$
  - Input space: $E[\boldsymbol{z}\boldsymbol{z}^H] = \boldsymbol{R}_z$ → Output space: $E[\boldsymbol{y}\boldsymbol{y}^H] = \boldsymbol{I}$

$$E[\boldsymbol{y}\boldsymbol{y}^H] = E[\boldsymbol{W}\boldsymbol{z}\boldsymbol{z}^H\boldsymbol{W}^H] = \boldsymbol{W}E[\boldsymbol{z}\boldsymbol{z}^H]\boldsymbol{W}^H = \boldsymbol{W}\boldsymbol{R}_z\boldsymbol{W}^H$$

If $\boldsymbol{W} = \boldsymbol{\Lambda}^{-\frac{1}{2}}\boldsymbol{E}^H$, $E[\boldsymbol{y}\boldsymbol{y}^H] = \boldsymbol{\Lambda}^{-\frac{1}{2}}\boldsymbol{E}^H\boldsymbol{R}_z\boldsymbol{E}\boldsymbol{\Lambda}^{-\frac{1}{2}} = \boldsymbol{\Lambda}^{-\frac{1}{2}}\boldsymbol{\Lambda}\boldsymbol{\Lambda}^{-\frac{1}{2}} = \boldsymbol{I}$

Scaling ($\boldsymbol{\Lambda}^{-\frac{1}{2}}$) Transform ($\boldsymbol{E}^H$)

Eigenvalue decomposition: $\boldsymbol{R}_z = \boldsymbol{E}\boldsymbol{\Lambda}\boldsymbol{E}^H$

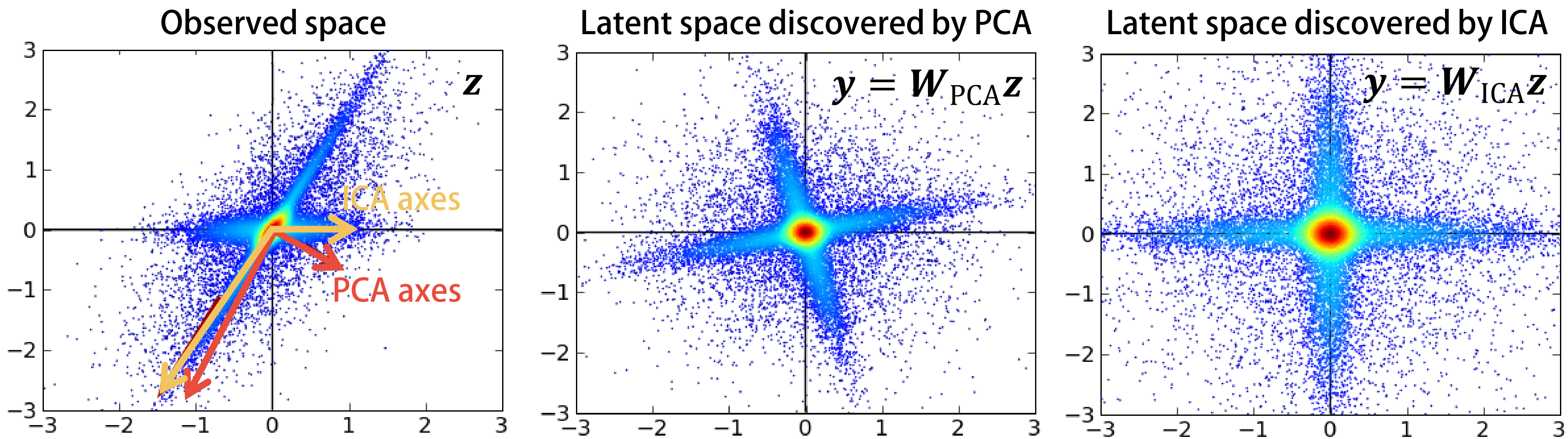


PCA and ICA

- PCA achieves second-order decorrelation
  - The dimensions of a latent space are diagonal
- ICA achieves higher-order decorrelation
  - The dimensions of a latent space are independent

Sufficient condition

$$\begin{bmatrix} y_1 \\ y_2 \end{bmatrix} = \begin{bmatrix} w_{11} & w_{12} \\ w_{21} & w_{22} \end{bmatrix} \begin{bmatrix} z_1 \\ z_2 \end{bmatrix} = \boldsymbol{w}_1 z_1 + \boldsymbol{w}_2 z_2$$



PCA and ICA

- PCA can be used as preprocessing of ICA
  - ICA filter $\boldsymbol{W}_{\text{ICA}}$ becomes unitary after performing PCA

$\boldsymbol{z}$ Observed vector → $\boldsymbol{y} = \boldsymbol{W}_{\text{PCA}}\boldsymbol{z}$ → $\boldsymbol{y}$ Output vector

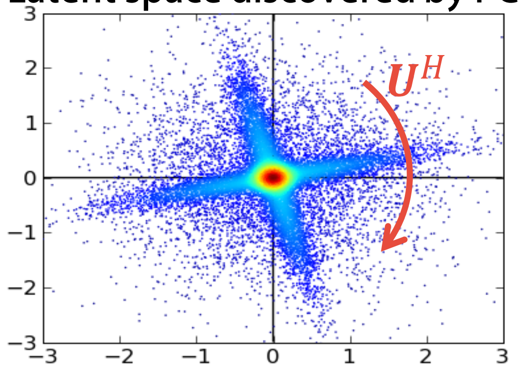
The requirement of PCA: $E[\boldsymbol{y}\boldsymbol{y}^H] = \boldsymbol{W}_{\text{PCA}}E[\boldsymbol{z}\boldsymbol{z}^H]\boldsymbol{W}_{\text{PCA}}^H = \boldsymbol{I}$

If we multiply any unitary matrix $\boldsymbol{U}^H$ $(\boldsymbol{U}^H\boldsymbol{U} = \boldsymbol{I}, \boldsymbol{W}_{\text{PCA}} \leftarrow \boldsymbol{U}^H\boldsymbol{W}_{\text{PCA}})$

$$\boldsymbol{y} = \boldsymbol{U}^H\boldsymbol{W}_{\text{PCA}}\boldsymbol{z} \longrightarrow E[\boldsymbol{y}\boldsymbol{y}^H] = \boldsymbol{U}^H\boldsymbol{W}_{\text{PCA}}E[\boldsymbol{z}\boldsymbol{z}^H]\boldsymbol{W}_{\text{PCA}}^H\boldsymbol{U} = \boldsymbol{U}^H\boldsymbol{U} = \boldsymbol{I}$$

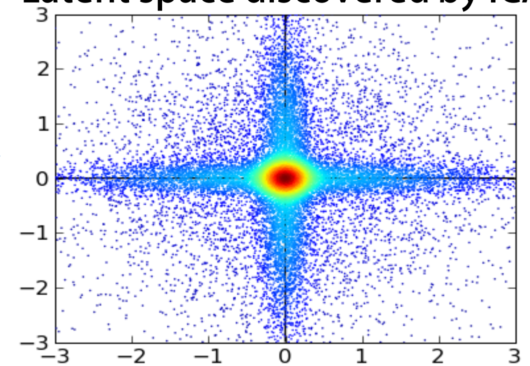
Latent space discovered by PCA

$\boldsymbol{y}' = \boldsymbol{W}_{\text{PCA}}\boldsymbol{z}$



$\boldsymbol{y} = \boldsymbol{U}^H\boldsymbol{y}'$

Latent space discovered by ICA



Cost Function

- Make the dimensions of a latent spaces independent
  - Minimize the KL divergence between $p(\boldsymbol{y})$ and $\prod_{i=1}^{N} p(y_i)$
    - If the dimensions of $\boldsymbol{y}$ are independent, $p(\boldsymbol{y}) = \prod_{i=1}^{N} p(y_i)$
    - We aim to make $p(\boldsymbol{y})$ as close to $\prod_{i=1}^{N} p(y_i)$ as possible

$$
\begin{aligned}
D_{KL}\left(p(\boldsymbol{y}) \middle\| \prod_{i=1}^{N} p(y_i)\right) &= \int p(\boldsymbol{y}) \log \frac{p(\boldsymbol{y})}{\prod_{i=1}^{N} p(y_i)} d\boldsymbol{y} \\
&= -\int p(\boldsymbol{y}) \log p(\boldsymbol{y})\, d\boldsymbol{y} + \sum_{i=1}^{N} \int p(y_i) \log p(y_i)\, dy_i \\
&= -H(\boldsymbol{y}) + \sum_{i=1}^{N} H(y_i)
\end{aligned}
$$

$$\boldsymbol{y} = \boldsymbol{W}\boldsymbol{z} \longrightarrow H(\boldsymbol{y}) = H(\boldsymbol{z}) + \log|\det(\boldsymbol{W})|$$

$$D_{KL} = -H(\boldsymbol{z}) - \log|\det(\boldsymbol{W})| - \sum_{i=1}^{N} E[\log p(y_i)]$$

Natural Gradient Algorithm

- Minimize the cost function by using a gradient method

Cost function

$$D_{KL} = -H(\boldsymbol{z}) - \log|\det(\boldsymbol{W})| - \sum_{i=1}^{N} E[\log p(y_i)]$$

$$\frac{\partial}{\partial w_{ij}} \sum_{i=1}^{N} E[\log p(y_i)] = E\left[\frac{\partial \log p(y_i)}{\partial y_i} \frac{\partial y_i}{\partial w_{ij}}\right] = E[-\varphi(y_i) z_j]$$

Gradient

$$\frac{\partial D_{KL}}{\partial \boldsymbol{W}} = -\boldsymbol{W}^{-H} + E[\boldsymbol{\varphi}(\boldsymbol{y})\boldsymbol{z}^H] = \left(\boldsymbol{I} - E[\boldsymbol{\varphi}(\boldsymbol{y})\boldsymbol{y}^H]\right)\boldsymbol{W}^{-H}$$

Score function

$$\varphi(y_i) = -\frac{\partial \log p(y_i)}{\partial y_i}$$

$$\boldsymbol{\varphi}(\boldsymbol{y}) = [\varphi(y_1), \cdots, \varphi(y_N)]^T$$

Natural gradient

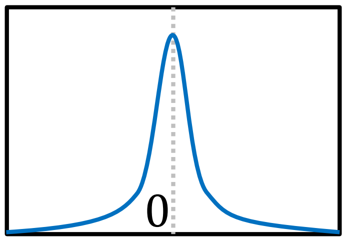
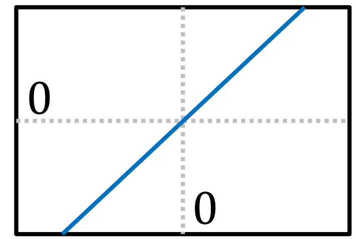
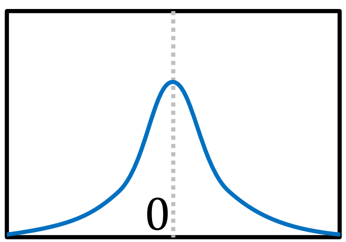
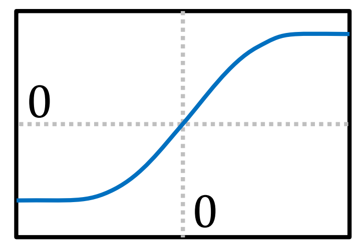
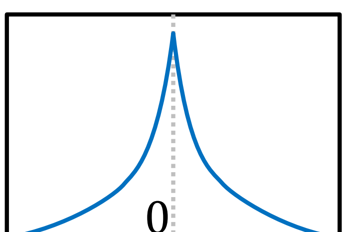
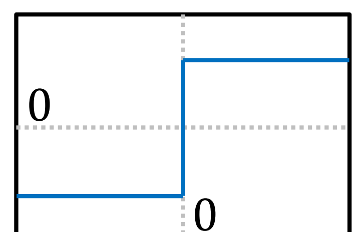
$$\frac{\partial D_{KL}}{\partial \boldsymbol{W}} \boldsymbol{W}^H \boldsymbol{W} = \left(\boldsymbol{I} - E[\boldsymbol{\varphi}(\boldsymbol{y})\boldsymbol{y}^H]\right)\boldsymbol{W}$$

Updating formula

$$\boldsymbol{W}_{t+1} = \boldsymbol{W}_t + \eta\left(\boldsymbol{I} - E[\boldsymbol{\varphi}(\boldsymbol{y})\boldsymbol{y}^H]\right)\boldsymbol{W}_t$$

Score Functions

- A distribution of source signal $s \approx y$ is required

	$p(y)$	$\varphi(y)$
Gaussian	 $\frac{1}{\sqrt{2\pi\sigma^2}}\exp\left(-\frac{\lvert y\rvert^2}{2\sigma^2}\right)$	 $\frac{y}{\sigma^2}$
Hyperbolic cosine	 $\frac{1}{\pi\cosh\frac{y}{\sigma^2}}$	 $\tanh\left(\frac{y}{\sigma^2}\right)$
Laplacian	 $\frac{1}{2\sigma}\exp\left(-\frac{\lvert y\rvert}{\sigma}\right)$	 $\frac{1}{\sigma}\operatorname{sgn}(y) = \frac{1}{\sigma}\frac{y}{\lvert y\rvert}$

Non-Gaussianity

- ICA assumes sound sources are NOT Gaussian distributed
  - The Gaussian distribution cannot be used as $p(y)$ in ICA

Score function: $\boldsymbol{\varphi}(\boldsymbol{y}) = [\varphi(y_1), \cdots, \varphi(y_N)]^T$

Updating formula: $\boldsymbol{W}_{t+1} = \boldsymbol{W}_t + \eta\left(\boldsymbol{I} - E[\boldsymbol{\varphi}(\boldsymbol{y})\boldsymbol{y}^H]\right)\boldsymbol{W}_t$

Gaussian case $\boldsymbol{\varphi}(\boldsymbol{y}) = \dfrac{\boldsymbol{y}}{\sigma^2}$ $E[\boldsymbol{\varphi}(\boldsymbol{y})\boldsymbol{y}^H] = \dfrac{1}{\sigma^2}E[\boldsymbol{y}\boldsymbol{y}^H] = \dfrac{1}{\sigma^2}\boldsymbol{R}_y$

→ The updating formula is depend on only second-order statistics

→ ICA reduces to PCA

Laplacian case $\varphi(y_i) = \dfrac{1}{\sigma}\dfrac{y_i}{|y_i|}$

→ Widely used for modeling speech and music signals

Maximum-Likelihood Estimation

- Estimate W such that $p(\boldsymbol{Z}|\boldsymbol{W})$ is maximized

ICA formulation: $\boldsymbol{y} = \boldsymbol{W}\boldsymbol{z}$

Independence of ICA outputs: $p(\boldsymbol{y}) = \prod_{i=1}^{N} p_i(y_i)$

$$p(\boldsymbol{z}) = |\det(\boldsymbol{W})| p(\boldsymbol{y})$$

$\boldsymbol{Z} = [\boldsymbol{z}_1, \cdots, \boldsymbol{z}_K]$ $\boldsymbol{z}_k$: observation at time k

$\boldsymbol{Y} = [\boldsymbol{y}_1, \cdots, \boldsymbol{y}_K]$ $\boldsymbol{y}_k$: ICA output at time k

$$p(\boldsymbol{Z}|\boldsymbol{W}) = \prod_{k=1}^{K} |\det(\boldsymbol{W})| \prod_{i=1}^{N} p_i(y_{i,k})$$

$$\frac{\partial p(\boldsymbol{Z}|\boldsymbol{W})}{\partial \boldsymbol{W}} = \left(\boldsymbol{I} - E[\boldsymbol{\varphi}(\boldsymbol{y})\boldsymbol{y}^H]\right)\boldsymbol{W}^{-H} \rightarrow \boldsymbol{0}$$

The same updating formulate is derived

Fast ICA

- ICA variant with a constraint $\boldsymbol{W}_{\mathrm{ICA}}^{H}\boldsymbol{W}_{\mathrm{ICA}} = \boldsymbol{I}$
  - PCA is used as preprocessing
  - Fewer iterations are required for convergence

Cost function

Restricted to be unitary

$$D_{KL} = -H(\boldsymbol{z}) - \log|\det(\boldsymbol{W})| - \sum_{i=1}^{N} E[\log p(y_i)] = -H(\boldsymbol{z}) - 1 - \sum_{i=1}^{N} E[\log p(y_i)]$$

Minimize

$$G(y_i) \equiv -\log p(y_i)$$

Optimization problem

$$\min_{W} \sum_{i=1}^{N} E[G(y_i)] \quad \text{subject to} \quad \boldsymbol{W}^{H}\boldsymbol{W} = \boldsymbol{I}$$

We have to design $G(y_i)$ such that $G(y_i) = -\log p(y_i) \approx -\log p(s_i)$

Learning Algorithm

- Choice of function $G(y_i)$

Example: generalized Laplacian: $p(y_i) \propto \exp\left(-\frac{\sqrt{|y_i|^2+\alpha}}{\sigma}\right)$ [Sawada 2004]

$$G(y_i) = \sqrt{|y_i|^2 + \alpha} \qquad g(y_i) = \frac{\partial G(y_i)}{\partial y_i} = \frac{y_i^*}{2\sqrt{|y_i|^2 + \alpha}}$$

$$g'(y_i) = \frac{\partial g(y_i)}{\partial y_i^*} = \frac{1}{2\sqrt{|y_i|^2 + \alpha}}\left(1 - \frac{1}{2}\frac{|y_i|^2}{|y_i|^2 + \alpha}\right)$$

- Updating formula of $\boldsymbol{W}$

$$\boldsymbol{y} = \boldsymbol{W}\boldsymbol{z} \qquad \boldsymbol{W} \equiv [\boldsymbol{w}_1, \boldsymbol{w}_2, \cdots, \boldsymbol{w}_M]^T$$

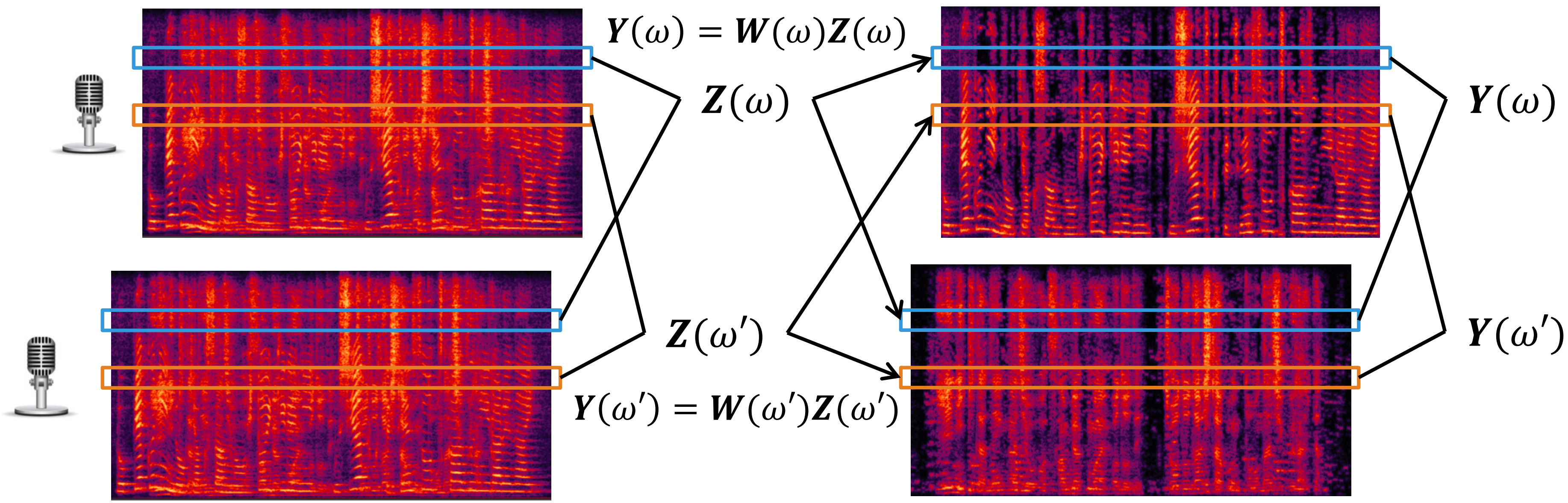
Update a filter: $\boldsymbol{W} \leftarrow E[g(y_i)\boldsymbol{z}] - E[g'(y_i)]\boldsymbol{w}_m$

Unitarize a filter: $\boldsymbol{W} \leftarrow \boldsymbol{W}(\boldsymbol{W}^H\boldsymbol{W})^{-\frac{1}{2}}$

Iterate until convergence

Problems

- Permutation ambiguity
  - The dimension order of Y cannot be determined uniquely
- Amplitude ambiguity
  - The dimension amplitude of Y cannot be determined uniquely



Solutions

- Solve permutation ambiguity
  - Focus on $\boldsymbol{y}$
    - Temporal power envelopes
  - Focus on $\boldsymbol{W}$
    - Directional patters of $\boldsymbol{W}$
    - Relative delay times from sources to microphones
    - Column vectors of $\boldsymbol{W}^{-1}$
- Solve amplitude ambiguity
  - Recover observed signals
    - Use the invserse of $\boldsymbol{W}$ for filtering each y_i

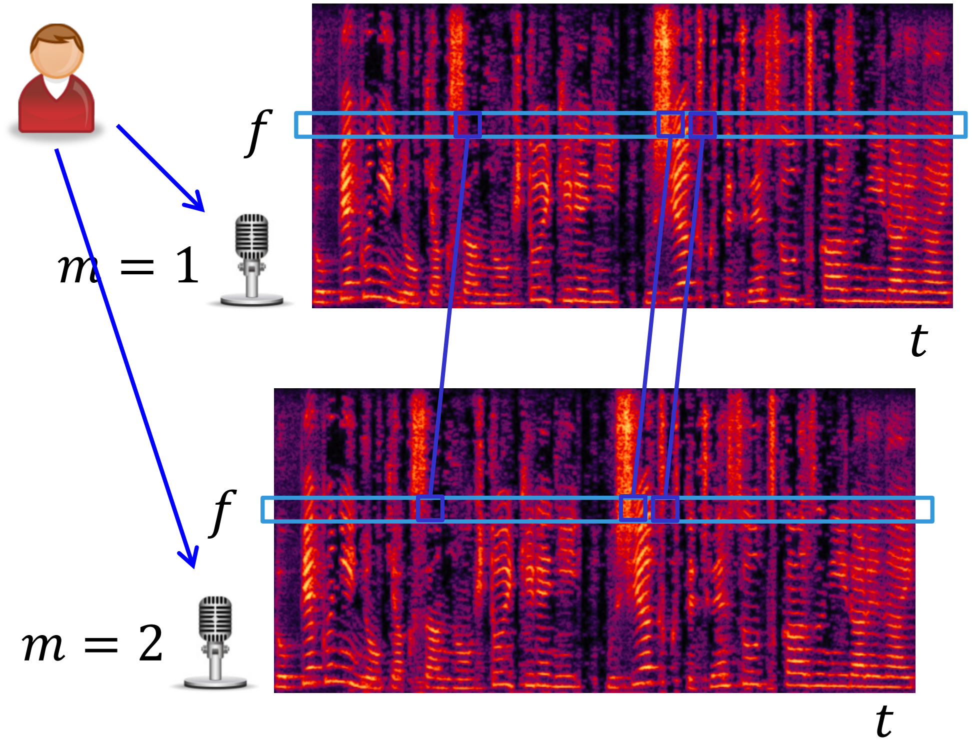
$$z_i = \boldsymbol{W}^{-1}[0, \cdots, 0, y_i, 0, \cdots, 0]^T$$

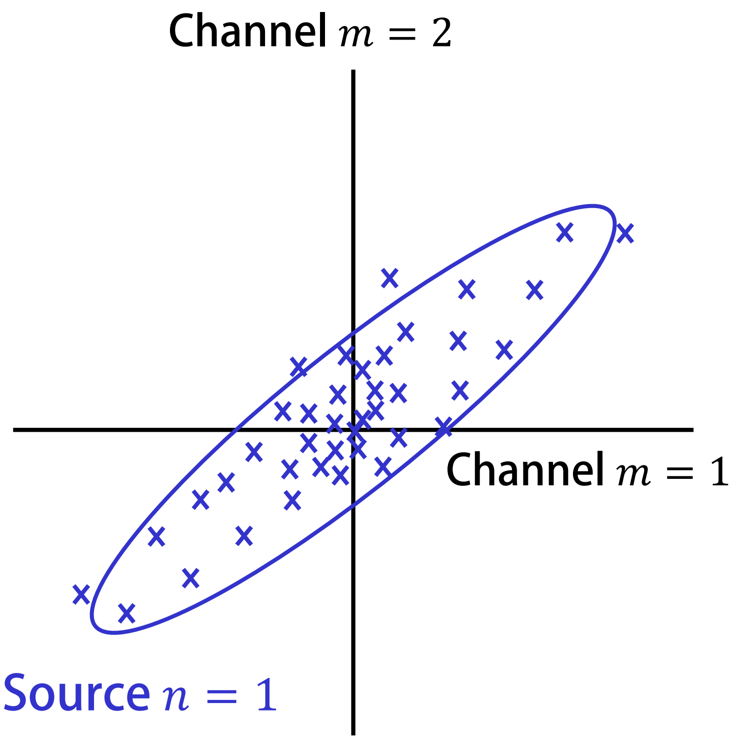
Nonlinear Time-Frequency Masking

Observation of Single Source

- The spectra of each source has a unique spatial property
  - The spectra are assumed to be Gaussian distributed

Observed data: $\boldsymbol{x}_{tf} = [x_{tf1}, x_{tf2}, \cdots, x_{tfM}]$

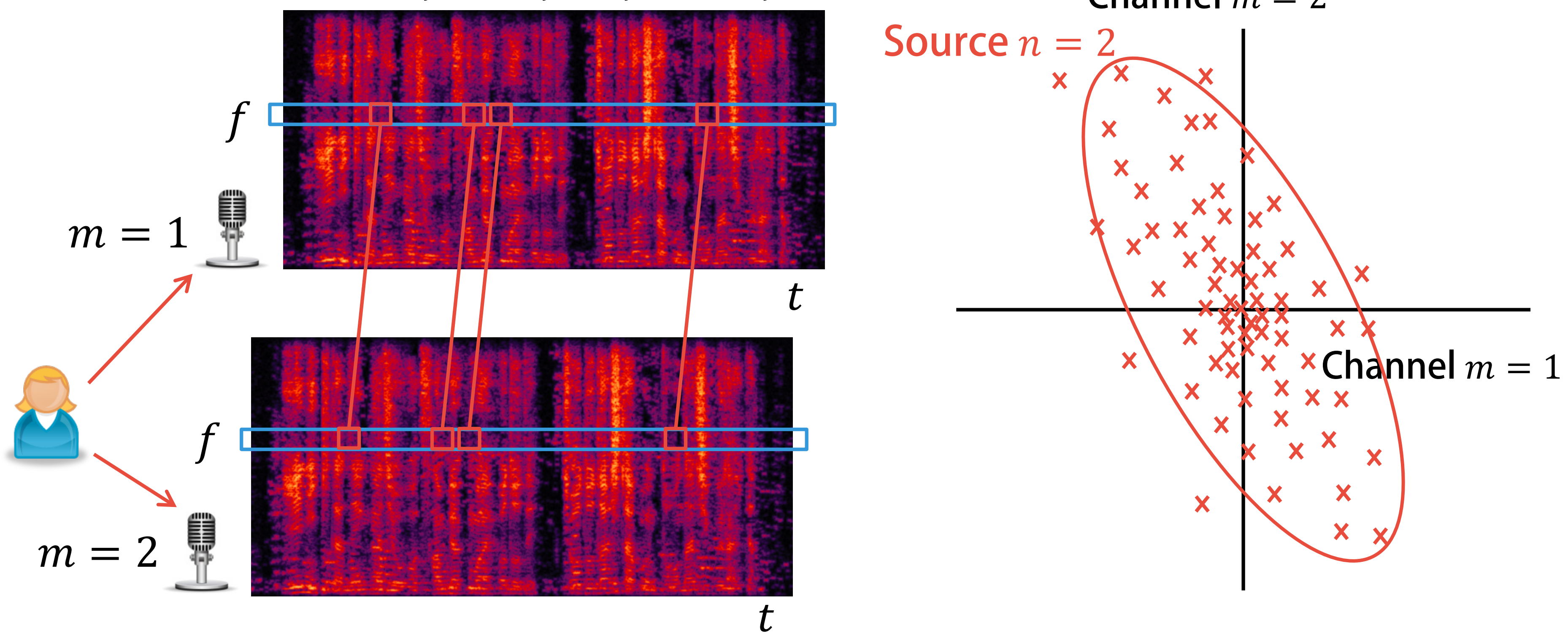




Observation of Single Source

- The spectra of each source has a unique spatial property
  - The spectra are assumed to be Gaussian distributed

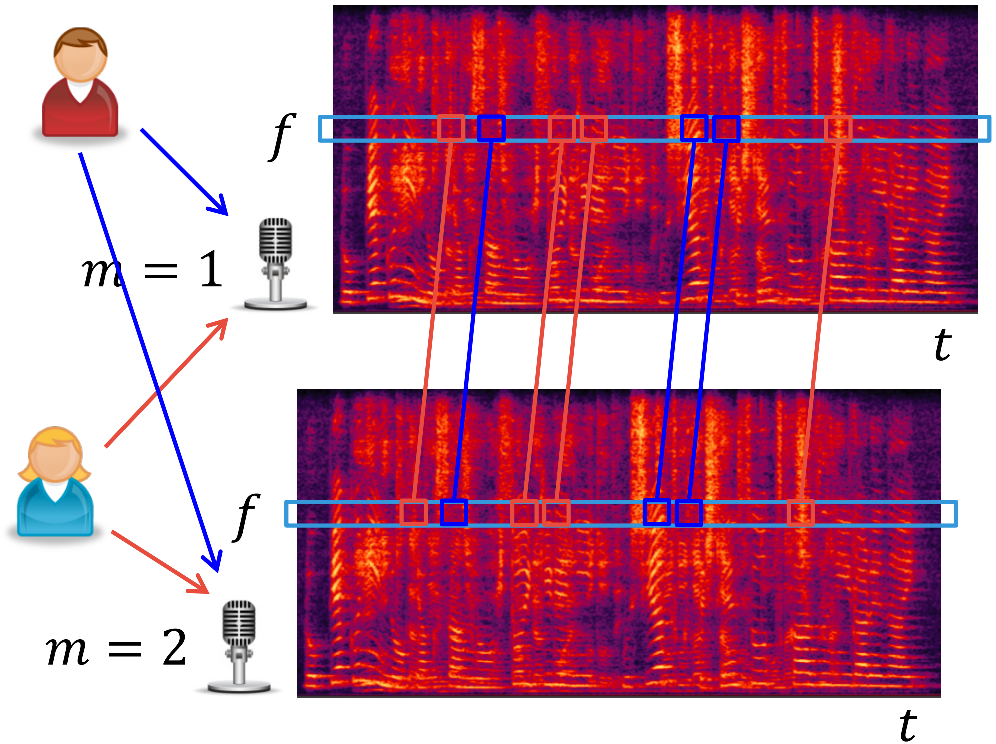
Observed data: $\boldsymbol{x}_{tf} = [x_{tf1}, x_{tf2}, \cdots, x_{tfM}]$

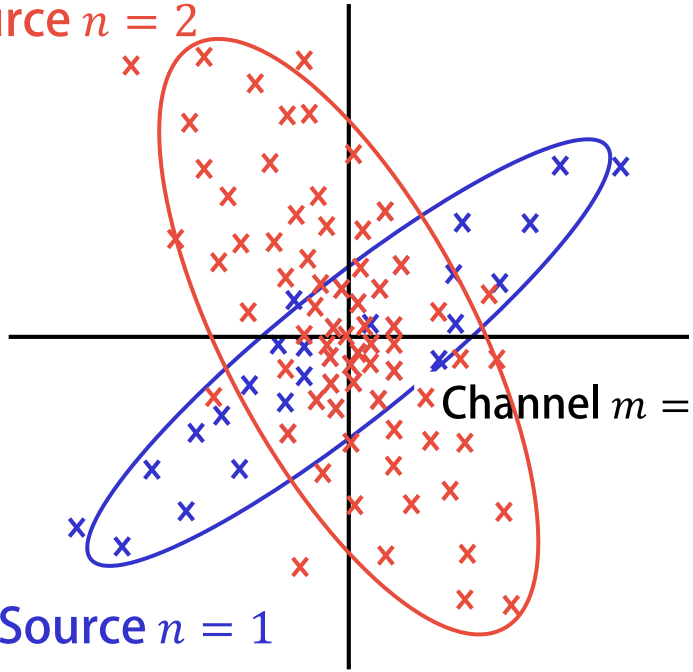


Observation of Multiple Sources

- The observed scatter plot is a mixture of spatial properties
  - Assume that source spectra are sparse (disjoint with each other)

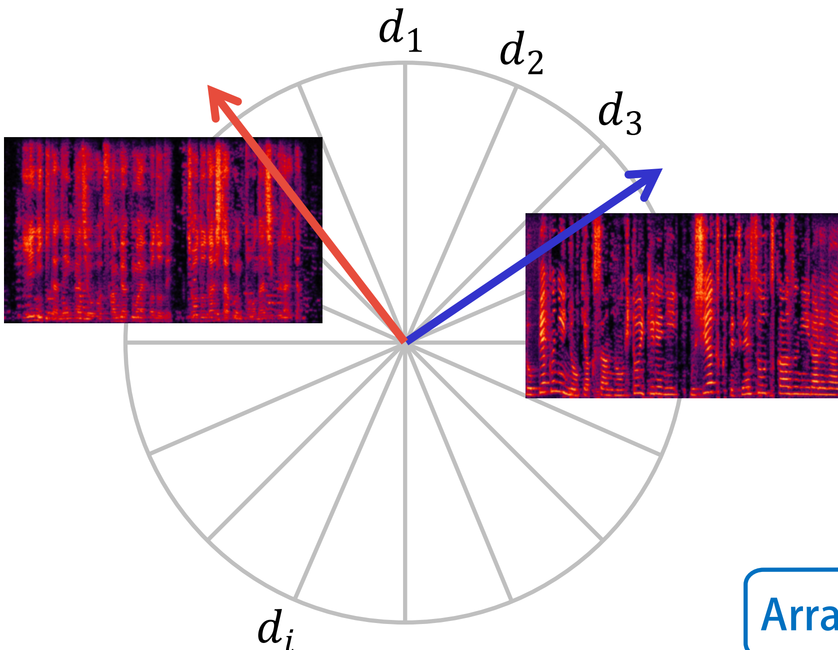
Observed data: $\boldsymbol{x}_{tf} = [x_{tf1}, x_{tf2}, \cdots, x_{tfM}]$





Time-Frequency Clustering

- Classify each frequency bin into one of sound sources
  - $z_{tf} = k$ indicates (time t, frequency f) is classified into source k
  - $\boldsymbol{H}_{fd}$: spatial correlation matrix for frequency f and direction d



Observation model [Duong 2010]

$$\boldsymbol{x}_{tf} \sim N_c\left(\boldsymbol{x}_{tf} \middle| \boldsymbol{0}, \left(\lambda_{tf} \boldsymbol{H}_{f d_{z_{tf}}}\right)^{-1}\right)$$

Source direction of time t and frequency f

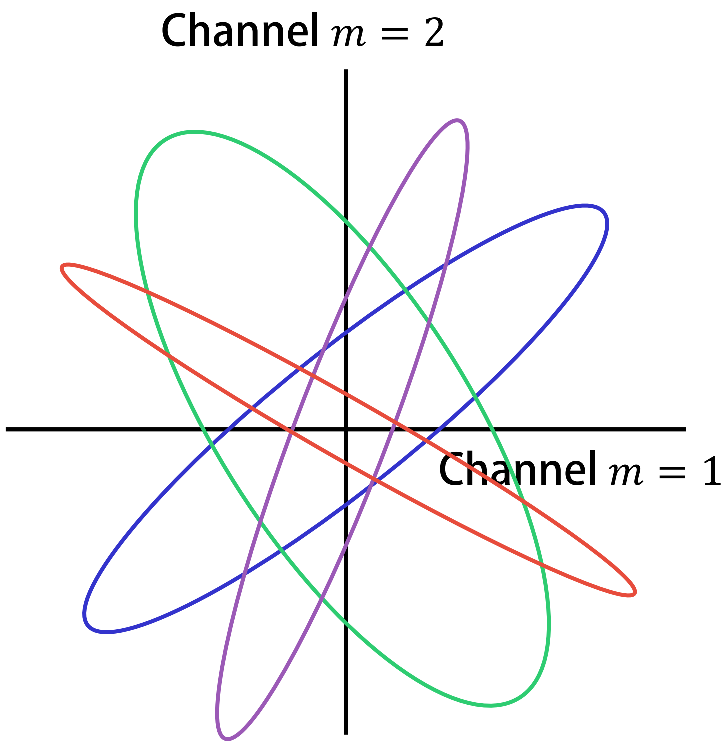
Bayesian formulation [Otsuka 2014]

$$\boldsymbol{H}_{fd} \sim W_c\left(\left(\boldsymbol{a}_{fd} \boldsymbol{a}_{fd}^H + \epsilon \boldsymbol{I}\right)^{-1}, \nu_0\right)$$

Array manifold vector for frequency f and direction d

Nonparametric Bayesian Extension

- Automatically estimate the number of sound sources
  - Assume that infinitely many sound sources exist in theory



Observation model [Duong 2010]

$$\boldsymbol{x}_{tf} \sim N_c\left(\boldsymbol{x}_{tf} \middle| \boldsymbol{0}, \left(\lambda_{tf} \boldsymbol{H}_{f d_{z_{tf}}}\right)^{-1}\right)$$

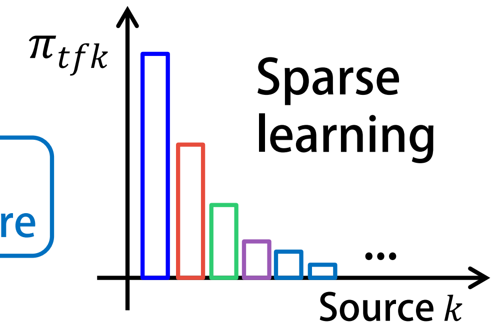
Source direction of time t and frequency f

Hierarchical Dirichlet process prior $(k \to \infty)$ [Otsuka 2014]

$$\boldsymbol{\pi}_{tf} \sim \mathrm{HDP}(\alpha, \gamma, \boldsymbol{\beta})$$

Concentration parameters — Base measure

$$z_{tf} \sim \mathrm{Categorical}(\boldsymbol{\pi}_{tf})$$



Advantages

- Simultaneous localization and separation
  - Improved performance of each task
    - Integration based on a probabilistic model
  - Automatic estimation of the number of sound sources
    - Nonparametric Bayesian formulation
  - Solving permutation problems
    - All frequency bins are simultaneously analyzed
- Various extensions feasible
  - Simultaneous dereverberation, localization, and separation [Otsuka 2014]
  - Analyzing moving sound sources [Otsuka 2014]
  - Real-time online inference (future work)

Assignment

- Questions
  - Explain delay-sum (DS) and minimum-variance (MV) beamforming methods using equations and why MV is better than DS.
  - Describe the relationships (differences) between PCA and ICA and how to estimate the parameters.
  - Report how microphone array processing is used in practice.
- How to submit
  - Submit a PDF file to "Assignment (Yoshii)" on PandA.
  - Deadline: 2018/01/30 23:59