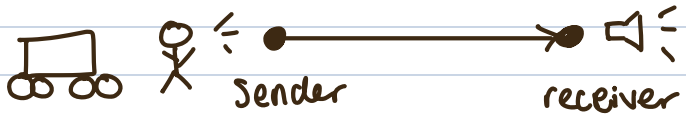
# Microphone Array Processing

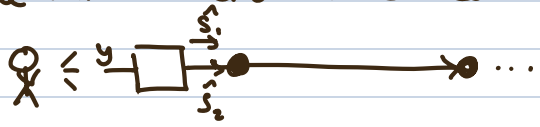
Project 2: Design & build a multi-mic speech enhancement / beamforming system for far-end noise reduction




far end speech enhancement

near end speech enhancement

Ex: Single mic far end noise reduction




## Speech Production (light notes due to past coursework)

- unvoiced /s/ (noisy) and voiced /a/ (periodic) and mixed /z/ and plosive /p/ (buildup of pressure)
- speech sound signal ≡ excitation signal
- formants (3-5 formants within Nyquist band)
  - ↳ resonance frequencies give rise to peaks in the overall spectrum
- Spectrograms → "see speech"
- Speech signals = vocal tract excitation + vocal tract filter
- vocal tract changes over time → time-varying
  - only short segments of speech can be assumed to have similar acoustic properties (non-stationary / short term stationary)
    - ↳ 20-30 ms time frames
- speech is stochastic
- max speech BW ~ 7-8 kHz

## Microphone Measurement Model

- Direct path $x[n] = a(d)s[n-\tau(d)]$
  - $s$: orig sound signal
  - $\tau(d)$: delayed by $\tau(d)$, time it takes to travel distance $d$
  - $a(d)$: scaling factor, depends on dist
  - ↳ sound goes to mic w/ some attenuation and delay

- reflections are modeled with a room impulse response

$$x[n] = (h * s)[n]$$

- $h$: room impulse response
- ↳ $h$ captures reflections

- direct path and early reflections determine intelligibility
- reverb (late refl.) degrade intelligibility

## Single Mic Model:

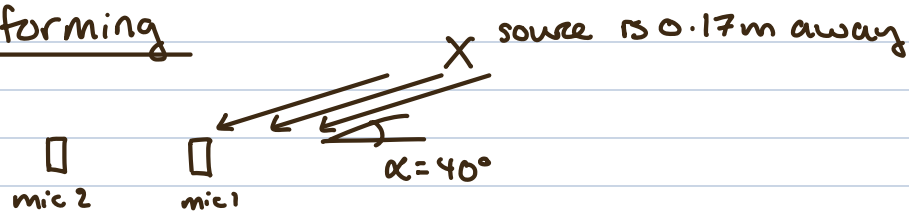
$$x[n] = \underbrace{\sum_{i=1}^{d} (h_e * s_i)[n]}_{\text{early}} + \underbrace{\sum_{i=1}^{d} (h_e * s_i)[n]}_{\text{late}} + n[n]$$

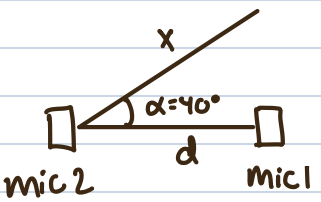
→ d point sources $S_i$
→ room impulse response $h_i$ from source position i to mic
→ $n[n]$: mic self noise & noise (e.g. late reverb)

Assume
- sources are additive, zero mean & uncorr, short term stationary

Early & late components correlated via source $S_i$

## Beamforming




$X = \frac{\tau}{f_s} c$

$\tau$: delay between mic 1 and 2

$\tau = \cos(\underset{\alpha=}{40°}) \frac{\underset{=0.17m}{d}}{c} f_s$

340 m/s, 8000 Hz

$\tau = 3.06$ samples, delay betw. mics
↳ non-int. → need to use freq. domain phase change to best descr.

Narrowband assumption?
↳ max delay across array is less than sampling period $T_s$
↳ in audio $T_s = 1/8000$ → meaning narrowband cannot apply for audio
↳ to use, need to process per frequency band
↳ beamformer resp. is freq. dependent
↳ phase shifts become freq. dependent

Spatial aliasing

$$d < \tfrac{1}{2}\lambda_{min} < \tfrac{1}{2}\frac{c}{\frac{1}{2}f_s} = \frac{c}{f_s}$$

✱ Narrowband can hold BUT ONLY if signal is processed in narrow frequency bands using the DFT
- delay T needs to be small enough to satisfy NBC in each f band

# Delay & Sum Beamformer (intro)

K: freq. index in discrete frequency domain

- 2 mics, one mic receives a delayed version
- sum coherently to get better SNR

$$\hat{S}(K,\ell) = \frac{X_1(K,\ell) + X_2(K,\ell)\, e^{j2\pi K\tau/N}}{2} = S(K,\ell) + \frac{N_1(K,\ell) + N_2(K,\ell) e^{j2\pi K\tau/N}}{2}$$

→ you have far field & near field effects to account for

**NEAR**

$$S(K,\ell) = s(K,\ell)\, a(d)\, e^{-j2\pi K\tau(d)/N}$$

- Source close to array center, each mic gets a different a(d) and phase difference $\tau$

**FAR**

$$s(K,\ell) = s(K,\ell)\, e^{-j2\pi K\tau(d)/N}$$

- only $\tau$

**FREE FIELD**

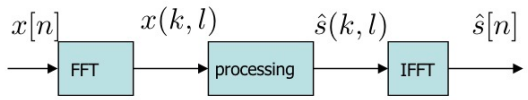
- only direct path, no refl.

## STFT

- time frames need to be short due to speech nature
- STFT makes conv. ≈ mult. & helps satisfy narrowband assumption

$$x(K,\ell) = \sum_{i=1}^{d} a_i(K,\ell)\, s_i(K,\ell) + n(K,\ell)$$

- for M mics $\underline{x}(K,\ell) = \sum_{i=1}^{d} \underline{a}_i(K,\ell)\, s_i(K,\ell) + \underline{n}(K,\ell)$




## Problem Formulation

$$\underline{X}(K,\ell) = \sum_{i=1}^{d} a_i(K,\ell)\, s_i(K,\ell) + n(K,\ell)$$

$$\mathbf{x}(k,l) = \underbrace{\mathbf{a}_1(k,l)s_1(k,l)}_{target} + \underbrace{\sum_{i=2}^{d} \mathbf{a}_i(k,l)s_i(k,l) + \mathbf{n}'(k,l)}_{interferers+noise}$$

$$= \mathbf{a}(k,l)s(k,l) + \mathbf{n}(k,l)$$

- Goal: Estimate $s(k,l)$ given $\mathbf{x}(k,l)$: e.g. $\hat{s}(k,l) = E[s(k,l)|\mathbf{x}(k,l)]$
- 1) Derive beamformers assuming $\mathbf{a}(k,l)$ is known.
- 2) estimation of $\mathbf{a}(k,l)$

$\underline{a}(K,\ell)$ : STFT of room impulse resp. per frequency stacked across mics

when normalized wrt ref mic → called RTF → shortens length of resp ! → est. target at ref. mic

(100 ms – 1 s)

Room impulse resp. >> frame size (20 ms)

# DELAY & SUM BEAMFORMER

our proj: multi mic / far end noise reduction

general idea: signal narrowband, time delay is f shift in Fourier domain, sum coherently after multiplying each sensor signal by a compensatory phase shift

$$\underline{x}(k,\ell) = S_1(k,\ell)\,\underline{a}(k,\ell) + \underline{n}(k,\ell)$$

$\underline{a}(k,\ell)$ ↓ models the ATF

$(k,\ell)$ freq., time

$$\hat{S}(k,\ell) = \underline{w}^H(k,\ell)\,\underline{x}(k,\ell)$$

$$\underline{a}(k,\ell) = \left[1,\ \frac{a_2 e^{-j2\pi k\tau_{2}/N}}{a_1},\ \ldots,\ \frac{a_M e^{-j2\pi k\tau_{M}/N}}{a_1}\right]^T \rightarrow \text{free \& near}$$

near & free field: $\underline{w}(k,\ell) = \dfrac{\underline{a}(k,\ell)}{\underline{a}^H(k,\ell)\,\underline{a}(k,\ell)}$

far & free field: $\underline{w}(k,\ell) = \frac{1}{M}\underline{a}(k,\ell)$ → I believe here, $\underline{a}(k,\ell) = \left[1,\ e^{-j2\pi k\tau_{2}/N},\ \ldots\right]$ because you don't account the $a(1)$ in far field

general case: $\underline{w}(k,\ell) = \dfrac{\underline{a}(k,\ell)}{\underline{a}^H(k,\ell)\,\underline{a}(k,\ell)}$

## Characteristics

- "preserves the target"
- no explicit knowledge of noise → good esp. if noise changes
- reduces noise var $\frac{1}{M} = \frac{1}{2^p} \Rightarrow -p\log_{10}(2) \approx -3p$ dB

# MVDR BEAMFORMER

Minimum Variance Distortionless Response
aka Capon beamformer

→ also exploits target location info
→ additionally exploits position of noise sources

- no change of mag or phase in dir of source
- min. variance of beamformer output in all other directions

cost function: $J(\underline{w}(k,\ell)) = \underline{w}^H(k,\ell)\,\underline{\underline{R}}_x(k,\ell)\,\underline{w}(k,\ell)$

↳ minimizes output power s.t. response in desired direction is 1 (this is called the distortionless constraint)

$$\min_{\underline{w}} \ \underline{w}^H(k,\ell)\,\underline{\underline{R}}_x(k,\ell)\,\underline{w}(k,\ell)$$

$$\text{s.t.} \quad \underline{w}(k,\ell)^H\,\underline{a}(k,\ell) = 1$$

* solve by taking deriv, set =0, plug in constraint and you get the following answer

$$\rightarrow \ \underline{w}(k,\ell) = \frac{(\underline{\underline{R}}_x(k,\ell))^{-1}\,\underline{a}(k,\ell)}{\underline{a}^H(k,\ell)(R_x(k,\ell))^{-1}\,\underline{a}(k,\ell)}$$

- You can also write this result by using $\underline{\underline{R}}_n(k,\ell)$

$$\underline{w}(k,\ell) = \frac{\underline{\underline{R}}_n^{-1}(k,\ell)\,\underline{a}(k,\ell)}{\underline{a}^H(k,\ell)\,\underline{\underline{R}}_n^{-1}(k,\ell)\,\underline{a}(k,\ell)} \qquad \underline{\underline{R}}_x(k,\ell) = \underline{\underline{R}}_n(k,\ell) + \underline{a}(k,\ell)\,a^H(k,\ell)\,\sigma_s^2(k,\ell)$$

Using MVDR Requires:

① $\underline{\underline{R}}_s(k,\ell)$ is rank 1

② target & noise are uncorrelated

③ target & noise are additive.

MVDR Spatially Uncorr. Noise: MVDR becomes the delay & sum beamformer

## Optimal Linear Multi Channel Wiener

↳ minimize MSE between the estimated signal and the true signal

Signal model: $\underline{x}(k,\ell) = s(k,\ell)\,\underline{a}(k,\ell) + \underline{n}(k,\ell)$

$$\min \ \mathbb{E}\left[\|s(k,\ell) - \underline{w}^H(k,\ell)\,\underline{x}(k,\ell)\|_2^2\right]$$

↳ $$\underline{w}(k,\ell) = \underline{\underline{R}}_x^{-1}(\ell)\,\sigma_{s,k}^2\,\underline{a}(k,\ell)$$

$$= \underbrace{\frac{\sigma_s^2(k,\ell)}{\sigma_s^2(k,\ell) + (\underline{a}^H(k,\ell)\,\underline{\underline{R}}_n^{-1}(k,\ell)\,\underline{a}(k,\ell))^{-1}}}_{\text{single ch w.F.}} \cdot \underbrace{\frac{\underline{\underline{R}}_n^{-1}(k,\ell)\,\underline{a}(k,\ell)}{\underline{a}^H(k,\ell)\,\underline{\underline{R}}_n^{-1}(k,\ell)\,\underline{a}(k,\ell)}}_{\text{MVDR}}$$

↓ post process MVDR filter to further reduce noise variance

↓ noise var set to remaining noise PSD after beamforming

↓ provides spatial filtering to supress noise & interference

## Sufficient Statistics

↳ a function of the data that captures all info you need to estimate a parameter of interest

$$T(\underline{x}(k,\ell)) = \underline{W}_{MVDR}^{H}(k,\ell)\,\underline{x}(k,\ell) = \frac{\underline{a}^{H}(k,\ell)\,\underline{\underline{R}}_{n}^{-1}(k,\ell)\,\underline{x}(k,\ell)}{\underline{a}^{H}(k,\ell)\,\underline{\underline{R}}_{n}^{-1}(k,\ell)\,\underline{a}(k,\ell)}$$

↓

the statistic is sufficient meaning using the statistic $T(\cdot)$ instead of the full data $x(k,\ell)$ does not lose any information about $s(k,\ell)$

## LCMV BEAMFORMER

Linearly Constrained Minimum Variance

- generalizes the MVDR by allowing multiple linear constraints
- generalize with $d$ linear constraints

$$\min_{\underline{w}(k,\ell)} \; \underline{w}^{H}(k,\ell)\,\underline{\underline{R}}_{x}(k,\ell)\,\underline{w}(k,\ell)$$

$$\text{s.t.} \quad \underline{w}^{H}(k,\ell)\,\underline{\underline{\Delta}}(k,\ell) = \underline{f}^{H}(k,\ell) \qquad \Delta \in \mathbb{C}^{M \times d}$$

when $d < M$, $\underline{w}(k,\ell) = \underline{\underline{R}}_{x}^{-1}(k,\ell)\,\underline{\underline{\Delta}}(k,\ell)\left(\underline{\underline{\Delta}}^{H}(k,\ell)\,\underline{\underline{R}}_{x}^{-1}(k,\ell)\,\underline{\underline{\Delta}}(k,\ell)\right)^{-1}\underline{f}(k,\ell)$

How can you use the multiple constraints?

→ steer zeros in direction of noise sources

→ maintain sig in certain dir.

✱ More constr = less deg of freedom to control noise reduction

## ESTIMATION OF THE A.T.F.

- all beamformers require an estimation of the ATF

$$x_m[n] = (h_m * s)[n]$$

$s$: point source

$m$: microphone

$h_m$: room impulse response

$x_m$: noise free source signal given at mth microphone

$$x_m(k,\ell) = a_m(k,\ell)\,s(k,\ell)$$

$a_m$: acoustic transfer function from source to m-th microphone

↳ temporal-frequency domain representation of the room impulse response

$\underline{x}(k,\ell) = \underline{a}(k,\ell)\, s(k,\ell)$ → vector form: collecting M microphone DFT coefficients in a vector

**Relative ATF**: $\underline{a}'(k,\ell) = \left[1, \frac{a_2(k,\ell)}{a_1(k,\ell)}, \dots\right]^T$ normalized wrt $a_1$

## Cross PSD Matricies

$$\mathbb{E}\left[\underbrace{\underline{x}(k,\ell)}_{\text{fourier domain}}\, \underline{x}(k,\ell)^H\right] = \mathbb{E}\left[\underbrace{\underline{s}(k,\ell)\, \underline{s}(k,\ell)^H}_{\text{describing how correlated signals are in the frequency domain}}\right] + \mathbb{E}\left[\underline{n}(k,\ell)\, \underline{n}(k,\ell)^H\right]$$

$$\underline{\underline{R}}_x(k,\ell) = \underline{\underline{R}}_s(k,\ell) + \underline{\underline{R}}_n(k,\ell)$$

**Example**: one point source

$\underline{s}(k,\ell) = \underline{a}(k,\ell)\, \underline{s}_1(k,\ell)$ $\quad \underline{a} \in \mathbb{C}^M$ : RTF

$\underline{\underline{R}}_s = \mathbb{E}\left[\underline{s}(k,\ell)\, \underline{s}(k,\ell)^H\right] = \sigma_{s_1}^2\, \underline{a}(k,\ell)\, \underline{a}(k,\ell)^H$

$\mathbb{E}\left[s_1(k,\ell)\, s_1(k,\ell)^H\right]$

$= \mathbb{E}\left[|s_1(k,\ell)|\right]^2$

= var of clean sig at ref mic 1

$\underline{\underline{R}}_s = 1$ in this case, should match up to # sources

★ when multiple sources, beamformers a funct. of $\underline{\underline{R}}_s$ ★

To solve beamforming
1. need a good estimate of $\underline{a}(k,\ell)$ (ATF / RTF)
2. if multiple sources, will need a good est. of $\underline{\underline{R}}_s$ → the beamforming eqns also need to be adapted for multiple sources

## Recall Eigenvalues:

$A = T \Lambda T^{-1}$, if A is hermitian $A = T \Lambda T^H$

## Estimate $\underline{a}(k,\ell)$ – No Noise — Using EVD

assume $\underline{\underline{R}}_s$ is perfectly known, $\underline{\underline{R}}_s = \sigma_{s_1}^2(k,\ell)\, \underline{a}(k,\ell)\, \underline{a}^H(k,\ell)$

$\underline{\underline{R}}_s = U \Lambda U^{-1} = U \Lambda U^H$

The scaled atf is given by $\underline{u}_1$: $\frac{\underline{a}(k,\ell)}{\|\underline{a}(k,\ell)\|} = \underline{u}_1$

## Estimate $\underline{a}(k,\ell)$ – Spatially white noise

**Using EVD**

Suppose $\underline{\underline{R}}_x(k,\ell) = R_s(k,\ell) + R_n(k,\ell)$
$= \sigma_s^2 \underline{a}(k,\ell)\underline{a}^H(k,\ell) + \sigma_n^2 \underline{\underline{I}}$

$\underline{\underline{R}}_x = U(\Lambda + \sigma_n^2 I)U^H$ ✱ only effects eigenvalues

so still estimate ATF by $\frac{\underline{a}(k,\ell)}{\|\underline{a}(k,\ell)\|} = \underline{u}_1$

✱ For spatially white noise, $R_x$ and $R_s$ share same eigenvectors $\underline{u}$.

Ultimately, you want to estimate $R_s$ b/c $R_s = U\Lambda U^H$ and $\vec{u}_1 = \frac{\underline{a}(k,\ell)}{\|\underline{a}(k,\ell)\|}$ so by getting $\hat{R}_s$, you can get $\underline{a}(k,\ell)$

**New problem formulation ⟹ estimate $R_s$**

## Estimate $R_s$ under spatially white noise

$\text{rank}(\underline{\underline{R}}_s) = r < M$

$$\underline{\underline{R}}_x = \begin{pmatrix} U_1 & U_2 \end{pmatrix} \begin{pmatrix} \Lambda + \sigma_n^2 I_r & \underline{0} \\ \underline{0} & \sigma_n^2 I_{M-r} \end{pmatrix} \begin{pmatrix} U_1^H \\ U_2^H \end{pmatrix}$$

$\underline{u}_1$ to $\underline{u}_r$ span speech + noise subspace

$\underline{u}_{r+1}$ to $\underline{u}_M$ span only noise subspace

so $\hat{\underline{\underline{R}}}_s = U_1(\Lambda_1 + \sigma_n^2 I_r)U_1^H$ → simplifies to

$\hat{\underline{\underline{R}}}_s = U_1 \Lambda_1 U_1^H$ → theoretically exact but irl noise introduces error

## Estimate $R_s$ under spatially colored data

**Using EVD**

① Pre-whiten data: $\underline{\underline{R}}_n^{1/2}$ : $\tilde{\underline{x}} = \underline{\underline{R}}_n^{1/2}\underline{x}$

② $R_{\tilde{x}} = \tilde{U}(\tilde{\Lambda} + I_M)\tilde{U}^H$, truncate $M-r$ smallest eig. to get $\tilde{U}_1, \tilde{\Lambda}_1, \tilde{U}_1^H = \tilde{R}_{\tilde{s}}$

③ Deconiten result: $\hat{\underline{\underline{R}}}_s = \underline{\underline{R}}_n^{1/2} \tilde{R}_{\tilde{s}} \underline{\underline{R}}_n^{1/2}$

✱ Note: using $R_n^{1/2}$ to dewhiten may result in a loss in accuracy

Therefore we do another method: generalized EVD

## GENERALIZED EVD FOR A.T.F. ESTIMATION

- hermitian matricies $\underline{\underline{A}}$ and $\underline{\underline{B}}$
- diagonalize both with $\underline{\underline{U}}$ → columns of $\underline{\underline{U}}$ are the generalized eigenvectors of $\underline{\underline{A}}$ and $\underline{\underline{B}}$
- the generalized eigenvalues and eigenvectors of $\underline{\underline{A}}, \underline{\underline{B}}$ are the ordinary eigenvalues and eigenvectors of $\underline{\underline{B}}^{-1}\underline{\underline{A}}$

Derivation

$$\left.\begin{aligned} U^H A U &= \Lambda_A \\ U^H B U &= \Lambda_B \end{aligned}\right\} \quad AU = BU\Lambda \qquad \Lambda = \Lambda_B^{-1}\Lambda_A$$

→ the generalized eigenvalues in $\Lambda$ are $\lambda_i = \frac{a_i}{b_i}$

$$\underline{\underline{R}}_x = \underline{\underline{U}}^{-H}(\underline{\underline{\Lambda}} + \underline{\underline{I}}_M)\underline{\underline{U}}^{-1}$$ → then assume $\text{rank}(R_s) = r < M$

→ then partition,

$$R_x = (Q_1\ Q_2)\begin{pmatrix} \underline{\Lambda}_1 + \underline{I}_r & \underline{0} \\ \underline{0} & \underline{I}_{M-r} \end{pmatrix}\begin{pmatrix} \underline{Q}_1^H \\ \underline{Q}_2^H \end{pmatrix} \qquad \begin{aligned} Q_1 &\in \mathbb{C}^{M\times r} \\ Q_2 &\in \mathbb{C}^{M\times(M-r)} \end{aligned}$$

→ Estimate $R_s$ from GEVD of $R_x$ : $\hat{R}_s = Q_1(\Lambda_1 + I_r)Q_1^H$

→ Simplify to (when one source) : $\hat{R}_s = Q_1 \Lambda_1 Q_1^H$

If rank $R_s = 1$, then ATF can be obtained by selecting $q_1$ the principle generalized eigenvector between $R_s(n,\ell)$ and $R_n(n,\ell)$

## GEVD application to Beamforming

One way to write $R_x$ is, $R_x = Q_1(\Lambda_1 + I_r)Q_1^H + Q_2Q_2^H$

Recall beamformer takes linear combos of microphone signals ( $\hat{S} = W^H X$)

Thus, $\hat{R}_s = W^H R_x W = W^H Q_1(\Lambda_1 + I_r)Q_1^H W + W^H Q_2 Q_2^H W$

✱ A good beamformer can be expressed as linear combos of $U_1$ cols

$W = U_1 b$ , $b \in \mathbb{C}^r$

# Performance Metrics

1. output SNR
2. MSE
3. noise reduction
4. speech distortion

## Output SNR

$$SNR_{out}(\underline{w}) = \frac{\underline{w}^H \underline{\underline{R}}_s \underline{w}}{\underline{w}^H \underline{\underline{R}}_n \underline{w}}$$

Notably,

$\nabla_{w^H} SNR_{out}(\underline{w}) = 0$ meaning $\underline{w}$ is a stationary point (as per operator theory)

So if we consider $\underline{w}$ to be a generalized eigenvector, then,

↳ $\underline{w} = \underline{w}_1$ maximizes output SNR

$$\frac{\underline{w}^H \underline{\underline{R}}_s \underline{w}}{\underline{w}^H \underline{\underline{R}}_n \underline{w}} = \lambda \quad \text{where } \lambda \text{ is a generalized eigenvalue}$$

This lets us say,

$$SNR_{out}(\underline{w}) \leq \max_{\underline{w}} \frac{\underline{w}^H \underline{\underline{R}}_s \underline{w}}{\underline{w}^H \underline{\underline{R}}_n \underline{w}} = \lambda_1$$

★ Note: result independent of $r = \text{rank}(R_s)$

## MSE

$$\mathbb{E}\left[|\underline{w}^H \underline{x} - s_1|^2\right] = \mathbb{E}\left[|\underline{w}^H \underline{s} + \underline{w}^H \underline{n} - s_1|^2\right]$$

$$= \underbrace{\mathbb{E}\left[|\underline{w}^H \underline{s} - s_1|^2\right]}_{\text{signal distortion}} + \underbrace{\mathbb{E}\left[|\underline{w}^H \underline{n}|^2\right]}_{\text{residual noise var}}$$

$$\min \mathbb{E}\left[|\underline{w}^H \underline{s} - s_1|^2\right]$$
$$\text{s.t. } \mathbb{E}\left[|\underline{w}^H \underline{n}|^2\right] \leq c \qquad 0 \leq c \leq \sigma_{n_1}^2 \rightarrow \text{noise at ref. mic before beamforming}$$

Solution: $\underline{w}^* = \underline{\underline{U}}\, b^*$

$$= \underline{\underline{U}} (\underline{\underline{\Lambda}} + \mu \underline{\underline{I}}_M)^{-1} \underline{\underline{U}}^H \underline{\underline{R}}_s \underline{e}_1$$

Lagrange mult. $\mu \geq 0$ chosen such that $b^H b = 0$

→ when we have $\text{rank}(R_s) = r < M$, then $R_s = Q_1 \Lambda_1 Q_1^H$

so

$$w^* = \underline{\underline{U}}_1 (\underline{\underline{\Lambda}}_1 + \mu \underline{\underline{I}}_r)^{-1} \underline{\underline{\Lambda}}_1 \underline{\underline{Q}}_1^H \underline{e}_1$$

∴ MMSE opt. beamformer can be expressed as lin combo of $U_1$ columns

→ another math simplification,

$$w^* = (\underline{\underline{R}}_s + \mu \underline{\underline{R}}_n)^{-1} \underline{\underline{R}}_s \underline{e}_1$$

↳ signal-distortion weighted (SDW) Wiener filter

* $\mu$ is a "trade off param" controlling signal dist. & noise reduction

* when $\mu = 1$ → multi-ch W.F. $W_{MWF} = \underline{\underline{R}}_x^{-1} \underline{\underline{R}}_s \underline{e}_1 = \sigma_{s_1}^2 \underline{\underline{R}}_x^{-1} \underline{a}$

* when $\mu = 0$ and rank r, $w^* = U Q_1^H e_1 = U_1 Q_1 e_1$ → the MVDR beamformer
↓response distortionless

↳ when $r = 1$, $w^* = \lambda_1^{-1} \sigma_{s_1}^2 \underline{\underline{R}}_n^{-1} \underline{a}$

and doing more simp., $w^* = \dfrac{\underline{\underline{R}}_n^{-1} \underline{a}}{\underline{a}^H \underline{\underline{R}}_n^{-1} \underline{a}}$

* General Case again but with $r = 1$,

$$w^* = (\underline{\underline{R}}_s + \mu \underline{\underline{R}}_n)^{-1} \underline{\underline{R}}_s \underline{e}_1$$

$$= \underbrace{\frac{\sigma_{s_1}^2}{\sigma_{s_1}^2 + \mu (\underline{a}^H \underline{\underline{R}}_n^{-1} \underline{a})^{-1}}}_{\text{MVDR}} \cdot \underbrace{\frac{\underline{\underline{R}}_n^{-1} \underline{a}}{\underline{a}^H \underline{\underline{R}}_n^{-1} \underline{a}}}_{\text{single ch W.F.}}$$

* SDW MWF can be MVDR and then s.c.W.F.

## ESTIMATE A.T.F. VIA CRAMER RAO BOUND → presented by PhD student

* using C.R. L.B. to do ATF estimation
* seems like a side quest, a potential option of further implementation
  ↳ a challenge
    ↳ will skip details for now

## ESTIMATE A.T.F. WITH INTER FREQUENCY CORRELATION

- improve acoustic parameter est. by exploiting hidden corr. across freq.

↳ more state of the art research → same, challenge

## Projects: Multi-Mic Noise Reduction (class notes)

↳ at some point you need to estimate A.T.F. → do via EVD or GEVD

↓

! only accurate when $R_x$ and $R_n$ known! estimation errors severely effect results

* Grad project: combine DL / GEVD to estimate ATF

# Perception & Binaural SP

*Binaural typically refers to hearing aids and is out of scope of project

## Measuring intelligibility

- matrix test
- speech recognition threshold
- audiogram

SII : speech intel. index (weighted SNR to intelligibility)

STI, CSII, STOI, HASPI

## Hints for Project 2

signal model: $x_m[t] = (s_1 * h_{1,m})[t] + \sum_{p=2}^{P} (s_p * h_{p,m})[t]$ (we called n in the notes)

Constructing the noisy signal:

- For sources $s_p$ and microphone $m$: $x_m[t] = (s_1 * h_{1,m})[n] + \sum_{p=2}^{P}(s_p * h_{p,m})[t]$
- Processing using STFT (i.e., using short time frames of 20 ms): window and FFT the samples $x_m[\text{overlap}(l-1)+1 : \text{overlap}(l-1)+\text{frsize}]$

Estimating Correlation matrices: $\mathbf{R_n}(k,l) = E[\mathbf{n}(k,l)\mathbf{n}^H(k,l)]$ and $\mathbf{R_x}(k,l) = E[\mathbf{x}(k,l)\mathbf{x}^H(k,l)]$

- Assuming ergodicity (sources are spatially invariant) you can estimate $\mathbf{R_n}(k,l)$ e.g. as

$$\hat{\mathbf{R}}_\mathbf{n}(k,l) = \frac{1}{N}\sum_{p=l-M_1}^{l+M_2} \mathbf{n}(k,p)\mathbf{n}^H(k,p)$$

or as

$$\hat{\mathbf{R}}_\mathbf{n}(k,l) = \begin{cases} \hat{\mathbf{R}}_\mathbf{n}(k,l-1)\alpha + \mathbf{n}(k,l)\mathbf{n}^H(k,l)(1-\alpha) & \text{target not present} \\ \hat{\mathbf{R}}_\mathbf{n}(k,l-1) & \text{target is present} \end{cases}$$

$$\hat{\mathbf{R}}_\mathbf{x}(k,l) = \hat{\mathbf{R}}_\mathbf{x}(k,l-1)\alpha + \mathbf{x}(k,l)\mathbf{x}^H(k,l)(1-\alpha)$$

- How to know whether the target is present or not? Either cheat by using directly the mix of interferers, or build a detector.

# Project ToDos / Outline

## Part 1 : generate signals

- use audio files given + impulse responses

$$x_m[t] = (s_1 * h_{1,m})[t] + \sum_{p=2}^{P} (s_p * h_{p,m})[t]$$

m: can be arbitrary but let's say $m = 4$ → jk given 4 mics for now → enough to get much better SNR if beamforming correct

going to assume only one intended speaker per setup. means only one intended source to reconstruct

two clean speech signals given , microphone impulse resp given as mat file

Matlab file:

$s_1 * h_1$ → mic 1 ... → should be able to play them
↳ assuming $h_1$ corr. to $s_1$

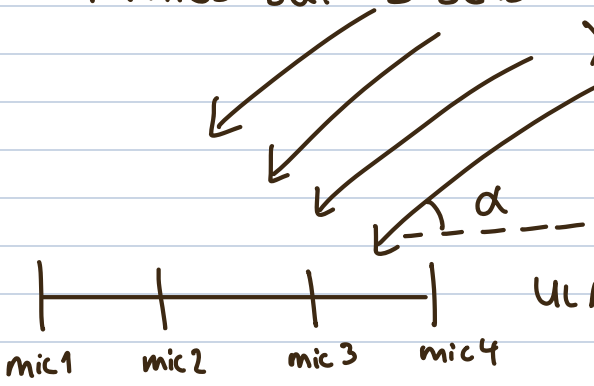
| | |
|---|---|
| h_inter1 | 4 x 400 |
| h_inter2 | 4 x 400 |
| h_inter3 | 4 x 400 |
| h_inter4 | 4 x 400 |
| h_target | 4 x 400 |

* let's see if we can plot these
convolve with sound signal to get signal at mic locations

→ assumption: sampled at the same sampling frequency as the speech signals
↳ 16k sampl. → 16 samples per ms

"4 mics" but "5 sets" → is h_target some sort of reference?

X source is a certain distance away




ULA? probably → yes, confirmed from fig 5 in paper

$$x_{mic1} = s_1 * h_{1,1} + \sum_{p=2}^{5} (s_p * h_{p,mic1})[t]$$

5 → because 5 h's

* P = 5 would imply 5 sources but two clean speech signals gotten?
↓
you can make any combination of the 5 sources

* In report also probably need to write something to defend the narrowband assumption

# About the Speech files given:

- 16-bit sampled at 16kHz
- Signal BW is at maximum $\frac{16k}{2} = 8kHz \rightarrow$ wideband speech

| | | | |
|---|---|---|---|
| s1 : clean speech 1 . wav | 36.1 sec | one female speaker |
| s2 : clean speech 2 . wav | 33.61 sec | 4 female, 4 male speakers |
| s3 : babble noise . wav | 37 sec | |
| s4 : artif nonstat noise . wav | 36.1 sec | |
| s5 : Speech shaped noise . wav | 90 sec | |

"constant interferers" → s4, s5
"nonst. interferers" → s2, s3

* s2 can have a direction so beamf. should work for that
* Start slow with only one clean, one noise
* make all same length
* can also start with even shorter segments to make data more workable at the start like 5s / 10s

* Need to also check the SNR of the created signals
  - ↳ negative SNRs may not be recoverable ... consider setting up a test with s1 pos. SNR then s1 negative SNR maybe

| Setup 1 | Setup 2 | Setup 3 |
|---|---|---|
| s1: clean 1 | s1: clean 1 + speech sh. noise 1 | s1: same as 2 |
| s2: clean 2 | s2: clean 2 | s2: clean 2 |
| s3: nonstat n | s3: nonstat n | s3: babble |
| s4: speech sh. #1 (split up) | s4: speech sh. #1 (split up) | s4: nonstat |
| s5: speech sh #2 | s5: speech sh #2 | s5: speech sh #2 |
| "s1 is only clean, this is ideal" "no babble" | "s1 is now noisy" "no babble" | "s1 noisy" "babble" |

} not enough "base cases"

IDEAL: clean 1, all other sources off

high SNR: 25 dB
med SNR: 15
low SNR: 5

Scen. 1 clean 1 + clean2(noise)
- high, (medium) low SNR

Scen. 2 clean 1 + babble(noise)
- H, (M), L SNR

Scen. 3 clean 1 + art. non. stat (noise)
- H (M) L SNR

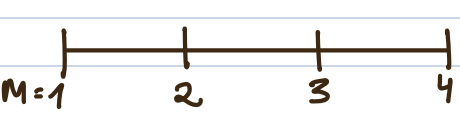
Scen. 4 clean 1 + speech shaped (noise)
- H (M) L SNR

} Go this to start, maybe just start w/ Med. to figure out what works
↳ Eval. on STOI

Scen. 5 (clean 1 + non. st.) + (clean 2 + speech-like noise)

Scen. 6 (clean 1) + (babble) + (clean 2)

& more later maybe




* Pay attn to SNR, it needs to be normalized (b/c like babble is too strong)

PART 1 STEPS:

⓪ Given $S_1$

① Generate noisy signals ($S_2$ thru $S_5$)

↳ you need to make $S_1$ evidently the intended speaker. This can be done by increasing the "volume" of $S_1$ and decr. the volume of the other clean speech signal which is not the intended signal

Eg.

$S_1$: clean speech 1 (high SNR) $S_3$: babble speech $S_5$: noise signal 2

$S_2$: clean speech 2 (low SNR) $S_4$: noise signal 1

↳ accomplish by cs 2 + noise_loud

② Generate $\underline{x}[t]$ array $\begin{bmatrix} x_1[t] \\ x_2[t] \\ x_3[t] \\ x_4[t] \end{bmatrix}$

↓

I think angles you can self choose

↓

probably just a good idea to sketch out setup so that you're always consistent

## Part 2 - Estimate the ATF

Methods we have learned:

→ EVD

→ GEVD

→ Cramer Rao Bound (likely out of scope but can consider as a challenge)

→ inter Frequency Correlation ← also

* Due to the various setups in part 1 → just do GEVD
* This replaces the h files
  ↳ first you test beamformers with h then do GEVD which simulates real world

### Estimate $R_s$ under spatially colored data (safe to assume here) — Using EVD

① Pre-whiten data: $\underline{\underline{R}}_n^{1/2}$ : $\tilde{\underline{x}} = \underline{\underline{R}}_n^{1/2}\,\underline{x}$

② $R_{\tilde{x}} = \tilde{U}(\tilde{\Lambda} + I_M)\tilde{U}^H$, truncate $M-r$ smallest eig. to get $\tilde{U}_1, \tilde{\Lambda}, \tilde{U}_1^H = \hat{R}_{\tilde{s}}$

③ Dewhiten result: $\hat{\underline{\underline{R}}}_s = \underline{\underline{R}}_n^{1/2}\,\hat{R}_{\tilde{s}}\,\underline{\underline{R}}_n^{1/2}$

* Note: using $R_n^{1/2}$ to dewhiten may result in a loss in accuracy (why?)

Therefore we do another method: generalized EVD

| Gen Sig. use h files Beamformers Evaluate | Gen Sig. GEVD for ATF Beamformers Evaluate |
|---|---|

* Here you need the detector to get $R_n$ and $R_s$
* Should be easy for clean 1

## GENERALIZED EVD FOR A.T.F. ESTIMATION

- hermitian matricies $\underline{\underline{A}}$ and $\underline{\underline{B}}$
- diagonalize both with $\underline{\underline{U}}$ → columns of $\underline{\underline{U}}$ are the generalized eigenvectors of $\underline{\underline{A}}$ and $\underline{\underline{B}}$
- the generalized eigenvalues and eigenvectors of $\underline{\underline{A}}, \underline{\underline{B}}$ are the ordinary eigenvalues and eigenvectors of $\underline{\underline{B}}^{-1}\underline{\underline{A}}$

**Derivation**

$$\left.\begin{aligned} U^H A U &= \Lambda_A \\ U^H B U &= \Lambda_B \end{aligned}\right\} \quad AU = BU\Lambda \qquad \Lambda = \Lambda_B^{-1}\Lambda_A$$

→ the generalized eigenvalues in $\Lambda$ are $\lambda_i = \frac{a_i}{b_i}$

$$\underline{\underline{R}}_x = \underline{\underline{U}}^{-H}(\underline{\underline{\Lambda}} + \underline{\underline{I}}_M)\underline{\underline{U}}^{-1}$$ → then assume $\text{rank}(R_s) = r < M$

→ then partition,

$$R_x = (Q_1\ Q_2)\begin{pmatrix} \underline{\underline{\Lambda}}_1 + \underline{\underline{I}}_r & \underline{\underline{0}} \\ \underline{\underline{0}} & \underline{\underline{I}}_{M-r} \end{pmatrix}\begin{pmatrix} Q_1^H \\ Q_2^H \end{pmatrix} \qquad \begin{aligned} Q_1 &\in \mathbb{C}^{M\times r} \\ Q_2 &\in \mathbb{C}^{M\times(M-r)} \end{aligned}$$

→ Estimate $R_s$ from GEVD of $R_x$ : $\hat{R}_s = Q_1(\Lambda_1 + I_r)Q_1^H$

→ Simplify to (when one source) : $\hat{R}_s = Q_1 \Lambda_1 Q_1^H$

If $\text{rank}\ R_s = 1$, then ATF can be obtained by selecting $q_1$ the principle generalized eigenvector between $R_s(k,\ell)$ and $R_n(k,\ell)$

# Part 3 - Creation of the Beamformers

Objective: multi-mic for far end noise reduction.

## Beamformer Types We Have Learned in Class Which Are Applicable:

- delay & sum
- MVDR
- MCWF
- LCMV

→ These 3 only take spatial sep. into account
↳ need to do MMSE noise reduction after
↳ do w/ single channel WF
↳ but then MVDR + S.C.W.F. is ↳ equal to the lin. constr. minimum power beamf. already the MWF?? no?

- equally weights spatial and noise filtering
  - ↳ variation is the SDW-MWF → with param $\mu$
    - $\mu = 1$ : SDW-MWF becomes MWF
    - $\mu \to 0$ : SDW-MWF becomes MVDR
- if satisfies constr. (see notes) it becomes the max. SNR beamformer
- can direct almost a perfect null towards interference source if spatial cov. matrix is rank 1

★ Easiest, just do Single Ch WF as post-processing ↳ for noise reduction

- MVDR and LCMV are considered distortionless beamformers

→ Technically as per the instructions we only have to implement a single beamforming system.

→ However, we should probably implement the 4 learned in class and intelligently highlight their strengths / weaknesses

## DELAY & SUM BEAMFORMER

pros / comments
cons

Characteristics

- "preserves the target"
- no explicit knowledge of noise → good esp. if noise changes
- reduces noise var $\frac{1}{M} = \frac{1}{2^p} \Rightarrow -p\log_{10}(2) \approx -3p$ dB

## MVDR BEAMFORMER

Minimum Variance Distortionless Response
aka Capon beamformer

→ also exploits target location info
→ additionally exploits position of noise sources

- no change of mag or phase in dir of source
- min. variance of beamformer output in all other directions

cost function: $J(\underline{w}(k,\ell)) = \underline{w}^H(k,\ell)\,\underline{\underline{R}}_x(k,\ell)\,\underline{w}(k,\ell)$

↳ minimizes output power s.t. response in desired direction is 1 (this is called the distortionless constraint)

Using MVDR Requires:

① $\underline{\underline{R}}_s(k,\ell)$ is rank 1

② target & noise are uncorrelated

③ target & noise are additive

MVDR Spatially Uncorr. Noise: MVDR becomes the delay & sum beamformer

# Optimal Linear Multi Channel Wiener

↳ minimize MSE between the estimated signal and the true signal

Signal model: $\underline{x}(k,\ell) = s(k,\ell)\,\underline{a}(k,\ell) + \underline{n}(k,\ell)$

$$\min \; E\left[\|s(k,\ell) - \underline{w}^H(k,\ell)\,\underline{x}(k,\ell)\|_2^2\right]$$

↳

$$\underline{w}(k,\ell) = \underline{\underline{R}}_x^{-1}(\ell)\,\sigma_{s,k}^2\,\underline{a}(k,\ell)$$

$$= \underbrace{\frac{\sigma_s^2(k,\ell)}{\sigma_s^2(k,\ell) + \left(\underline{a}^H(k,\ell)\,\underline{\underline{R}}_n^{-1}(k,\ell)\,\underline{a}(k,\ell)\right)^{-1}}}_{\text{single ch w.F.}} \cdot \underbrace{\frac{\underline{\underline{R}}_n^{-1}(k,\ell)\,\underline{a}(k,\ell)}{\underline{a}^H(k,\ell)\,\underline{\underline{R}}_n^{-1}(k,\ell)\,\underline{a}(k,\ell)}}_{\text{MVDR}}$$

single ch w.F. → post process MVDR filter to further reduce noise variance → noise var set to remaining noise PSD after beamforming

MVDR → provides spatial filtering to supress noise & interference

## LCMV Beamformer

Linearly Constrained Minimum Variance

- generalizes the MVDR by allowing multiple linear constraints
- generalize with $d$ linear constraints

$$\min_{\underline{w}(k,\ell)} \; \underline{w}^H(k,\ell)\,\underline{\underline{R}}_x(k,\ell)\,\underline{w}(k,\ell)$$

$$\text{s.t.} \quad \underline{w}^H(k,\ell)\,\underline{\underline{\Lambda}}(k,\ell) = \underline{f}^H(k,\ell) \qquad \Lambda \in \mathbb{C}^{M \times d}$$

when $d < M$, $\underline{w}(k,\ell) = \underline{\underline{R}}_x^{-1}(k,\ell)\,\underline{\underline{\Lambda}}(k,\ell)\left(\underline{\underline{\Lambda}}^H(k,\ell)\,\underline{\underline{R}}_x^{-1}(k,\ell)\,\underline{\underline{\Lambda}}(k,\ell)\right)^{-1}\underline{f}(k,\ell)$

How can you use the multiple constraints?

→ steer zeros in direction of noise sources → requires knowing where noise sources coming from

→ maintain sig in certain dir.

✗ More constr = less deg of freedom to control noise reduction

Beamformers Notes from Anja (https://nl.mathworks.com/help/phased/ug/beamforming-concepts.html)

This specifically reviews pros and cons of the beamformers and discusses what functions to call in matlab

A lot of this is directly copied from the website so it needs to be be reformatted before including snippets

- General
  - Conventional beamforming techniques include delay-and-sum beamforming, phase-shift beamforming, subband beamforming, and filter-and-sum beamforming. These beamformers are similar because the weights and parameters that define the beampattern are fixed and do not depend on the array input data.
- Delay & Sum explanation
  - plane waves arriving at a linear array have a time delay that is a linear function of distance along the array. Delay-and-sum beamforming compensates for these delays by applying a reverse delay to each sensor. If the time delay is accurately computed, the signals from each sensor add constructively. When the signal is narrowband, time delay becomes a phase shift in the frequency domain and is implement by multiplying each sensor signal by a frequency-dependent compensatory phase shift. This algorithm is implemented in the phased.PhaseShiftBeamformer.
- Delay & Sum cons
  - Finding the compensating delay at each sensor requires accurate knowledge of the sensor locations and signal direction.
- Delay & Sum pros
  - Another advantage is its robustness against pointing errors and signal direction errors.
- Delay & Sum cons
  - A disadvantage is its broad main lobe which decreases resolution of closely spaced sources or targets. A second disadvantage is that it has large sidelobes that allow interference sources to leak into the main beam.
- General
  - Also have Optimal and Adaptive Beamforming — data dependent beamformers
- MVDR general
  - Optimal beamformers apply weights that are determined by optimizing some quantity. The MVDR beamformer determines the beamforming weights, w, by maximizing the signal-to-noise+interference ratio of the array output
- MVDR general
  - Because of the constraint, beamformer preserves the desired signal while minimizing contributions to the array output due to noise and interference. The MVDR beamformer is implemented in phased.MVDRBeamformer.
- MVDR pros
  - The beamformer incorporates the noise and interference into an optimal solution. The beamformer has higher spatial resolution than a conventional beamformer. The beamformer puts nulls in the direction of any interference sources. Sidelobes are smaller and smoother.
- MVDR cons
  - The MVDR beamformer is sensitive to errors in either the array parameters or arrival direction. The MVDR beamformer is susceptible to self-nulling. In addition, trying to use MVDR as an adaptive beamformer requires a matrix inversion every time the noise and interference statistics change. When there are many array elements, the inversion can be computationally expensive.
- MVDR general — becoming MPDR be careful
  - it often turns out that the noise is not separable from the signal and it is impossible to determine Rn. In that case, you can estimate a sample covariance matrix from the data. and minimizes w'Rxw instead. Minimizing this quantity leads to the minimum power distortionless response (MPDR) beamformer. If the data vector, x, contains the signal and the estimated data covariance matrix is perfect and the steering vector of the desired signal is known exactly, the MPDR

beamformer is equivalent to the MVDR beamformer. However, MPDR degrades more severely when Rx is estimated from insufficient data or the signal arrival vector is not known precisely.

- LCMV general
  - The LCMV beamformer is a generalization of MVDR beamforming and is implemented in phased.LCMVBeamformer and phased.TimeDelayLCMVBeamformer. There are several different approaches to specifying constraints such as amplitude and derivative constraints. You can, for example, specify weights that suppress interfering signals arriving from a particular direction while passing signals from a different direction without distortion.
- LCMV pros and cons
  - The advantages and disadvantages of the MVDR beamformer also apply to the LCMV beamformer.
- Improved LCMV / MVDR by recursive implementation (save computation)
  - While MVDR and LCMV are adaptive in principle, re-computation of the weights requires the inversion of a potentially large covariance matrix when the array has many elements. The Frost and generalized sidelobe cancelers are reformulations of LCMV that convert the constrained optimization into minimizing an unconstrained form and then compute the weights recursively. This approach removes any need to invert a covariance matrix.
    See phased.FrostBeamformer and phased.GSCBeamformer.

## PART 4 - EVALUATION

Evaluation Metrics (base SP & perception metrics)

- output SNR
- MMSE
- SII : speech intel. index (weighted SNR to intelligibility)
- STI, CSII, STOI, HASPI