Introduction to Array Processing

Olivier Besson

Institut Supérieur de l'Aéronautique et de l'Espace

ISAE-SUPAERO
1 Introduction

Multichannel processing
A simple example: array processing with two antennas

2 Array processing model

3 Beamforming

4 Source localization
Context of multichannel processing

- Multichannel processing involves processing measurement vectors
  \[ y(k) = \begin{bmatrix} y_1(k) & y_2(k) & \ldots & y_N(k) \end{bmatrix}^T \in \mathbb{C}^N \]
  which can be:
  
  - \( N \) samples collected (possibly at the same time) on \( N \) different sensors, for instance the temperature measured in \( N \) different locations, the blood pressure of \( N \) different persons, etc.
  
  - \( N \) samples taken on a single sensor over a time frame of \( NT_s \) where \( T_s \) is the sampling period. In radar \( N \) is the number of pulses sent by a radar each \( T_s \) seconds during a coherent processing interval.
  
- In this course, we focus on signals received on an array of \( N \) antennas placed at different locations with a view to enhance reception of signals coming from preferred directions.
A generic multichannel problem

• Ubiquitous problem: the vectors \( y(k) \) contain some information (usually buried in noise) one wants to retrieve.

• A classical multichannel problem is to detect/estimate a known signal \( a \) that would be present in \( y \) in addition to some noise \( n \), i.e., decide whether \( \alpha = 0 \) or \( \alpha \neq 0 \) from observation of \( y = \alpha a + n \).

• The simplest way is to design a linear filter \( w \) such that

\[
\sum_{n=1}^{N} w_n^* y_n = \begin{bmatrix} w_1^* & w_2^* & \cdots & w_N^* \end{bmatrix} \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_N \end{bmatrix} = w^H y \approx \alpha
\]

and possibly to test the energy \( |w^H y|^2 \).
Array processing with two antennas

- Consider two antennas receiving a signal $s(t)$ emitted by a source in the far-field.

$$
y_1(t) \approx As(t - t_0)
y_2(t) \approx As(t - t_0 - \Delta t)
s(t)
$$

- The time delay $\Delta t$ depends on the direction of arrival $\theta$ of $s(t)$ and on the relative (known) positions of the antennas:
  - if $\theta$ is known, one can obtain $s(t)$: spatial filtering (beamforming)
  - if one can estimate $\Delta t$ from $y_1(t)$ and $y_2(t)$, then $\theta$ follows: source localization.
Beamforming with two antennas

- For narrowband signals, a delay amounts to a phase shift. Hence

\[ y(t) = \begin{bmatrix} y_1(t) \\ y_2(t) \end{bmatrix} = As(t) \begin{bmatrix} 1 \\ e^{i\phi} \end{bmatrix} + \begin{bmatrix} n_1(t) \\ n_2(t) \end{bmatrix} \]

- Let us use a linear filter to estimate \( As(t) \). The output is

\[ w_1^*y_1(t) + w_2^*y_2(t) = As(t)[w_1^* + w_2^*e^{i\phi}] + [w_1^*n_1(t) + w_2^*n_2(t)] \]

- A good filter maximizes the output signal to noise ratio (SNR) which, if \( n_1(t) \) and \( n_2(t) \) are uncorrelated and have same power, is given by

\[
\text{SNR} = \frac{|w_1^* + w_2^*e^{i\phi}|^2 |A|^2 P_s}{|w_1|^2 + |w_2|^2 P_n}
\]

\[ \rightarrow \text{maximum for } w_2 = w_1 e^{i\phi}, \text{ so that } w^H y(t) \propto y_1(t) + y_2(t)e^{-i\phi}. \]
Array of sensors offer an additional dimension (space) which enables one, possibly in conjunction with temporal or frequency filtering, to perform spatial filtering of signals:

1. source separation
2. direction finding

Fields of application

1. radar, sonar (detection, target localization, anti-jamming)
2. communications (system capacity improvement, enhanced signals reception, spatial focusing of transmissions, interference mitigation)
Contents

1 Introduction

2 Array processing model
   Principle
   Multichannel receiver
   Source signals
   Signals received on the array
   Covariance matrix

3 Beamforming

4 Source localization
Arrays and waveforms

- The array performs **spatial sampling** of a wavefront impinging from direction \((\theta, \phi)\).
- Assumptions: homogeneous propagation medium, source in the far-field of the array \(\rightarrow\) plane wavefront.
Multichannel receiver

HF filter
thermal noise

\[ \text{Re}[\tilde{y}_n(t)] \]
\[ \text{Im}[\tilde{y}_n(t)] \]

\[ \cos(\omega c t) \]
\[ -\sin(\omega c t) \]

Lowpass filter

\[ \text{Re}[y_n(t)] \]
\[ \text{Im}[y_n(t)] \]

\[ y(kT_s) \]

ADC

snapshot
Signals (in the frequency domain)

\[ \tilde{X}(\omega) \]

\[ \tilde{H}(\omega) \]

\[ S(\omega) \]

\[ \tilde{H}(\omega) \]

- \( \omega_c \)
- \( -\omega_c \)

O. Besson (ISAE-SUPAERO)
## Signals and receiver

### Source signal (narrowband)

\[
\tilde{x}(t) = 2\text{Re} \left\{ s(t)e^{i\omega_c t} \right\} \\
\triangleq \text{Re} \left\{ \alpha(t)e^{i\phi(t)}e^{i\omega_c t} \right\} \\
= \alpha(t) \cos [\omega_c t + \phi(t)]
\]

\(\alpha(t)\) and \(\phi(t)\) stand for amplitude and phase of \(s(t)\), and have slow time-variations relative to \(f_c\).

### Channel response

Receive channel number \(n\) has impulse response \(\tilde{h}_n(t)\).
Model of received signals

- Signal received on $n$-th antenna

$$\ddot{y}_n(t) = \alpha \ddot{h}_n(t) \ast \ddot{x}(t - \tau_n) + \ddot{n}_n(t)$$

where $\tau_n$ is the propagation delay to $n$-th sensor.

- In frequency domain:

$$\tilde{Y}_n(\omega) = \alpha \tilde{H}_n(\omega) \tilde{X}(\omega)e^{-i\omega\tau_n} + \tilde{N}_n(\omega)$$

- After demodulation ($\omega \to \omega + \omega_c$) and lowpass filtering:

$$Y_n(\omega) = \alpha \tilde{H}_n(\omega + \omega_c)S(\omega)e^{-i(\omega + \omega_c)\tau_n} + \tilde{N}_n(\omega + \omega_c)$$

$$\simeq \alpha \tilde{H}_n(\omega_c)S(\omega)e^{-i\omega_c\tau_n} + \tilde{N}_n(\omega + \omega_c)$$
Model of received signals

- Taking the inverse Fourier transform $F^{-1}(Y_n(\omega))$ yields

$$y_n(t) \approx \alpha \tilde{H}_n(\omega_c)s(t)e^{-i\omega_c \tau_n} + n_n(t)$$

- The signal is then sampled (temporally) at rate $T_s$ to obtain the $N|K$ data matrix $Y = \begin{bmatrix} y(1) & y(2) & \ldots & y(K) \end{bmatrix}$:
Model of received signals

- The **snapshot** at time index $k$ writes

$$y(k) = \begin{bmatrix} y_1(kT_s) \\ y_2(kT_s) \\ \vdots \\ y_N(kT_s) \end{bmatrix} = \alpha \begin{bmatrix} \hat{H}_1(\omega_c)e^{-i\omega_c\tau_1} \\ \hat{H}_2(\omega_c)e^{-i\omega_c\tau_2} \\ \vdots \\ \hat{H}_N(\omega_c)e^{-i\omega_c\tau_N} \end{bmatrix} s(kT_s) + \begin{bmatrix} n_1(kT_s) \\ n_2(kT_s) \\ \vdots \\ n_N(kT_s) \end{bmatrix}$$

- Assuming all $\hat{H}_n(\omega_c)$ are identical and absorbing $\alpha$ and $\hat{H}_n(\omega_c)$ in $s(kT_s)$, we simply write

$$y(k) = a(\theta)s(k) + n(k)$$

where $a(\theta)$ is the vector of phase shifts, referred to as the **steering vector** since $\tau_n$ depends only on the directions(s) of arrival of the source.
Model of received signals

Snapshot at time index \( k \)

The snapshot received in the presence of \( P \) sources is given by

\[
y(k) = \sum_{p=1}^{P} a(\theta_p)s_p(k) + n(k)
\]

\[
= \begin{bmatrix} a(\theta_1) & \ldots & a(\theta_P) \end{bmatrix} \begin{bmatrix} s_1(k) \\ \vdots \\ s_P(k) \end{bmatrix} + n(k)
\]

\[
= A(\theta)s(k) + n(k)
\]
Steering vector

$$\tau_n = \frac{1}{c} \left[ x_n \cos \theta \cos \phi + y_n \cos \theta \sin \phi + z_n \sin \theta \right]$$

$$a_n(\theta, \phi) = e^{i \frac{2\pi}{\lambda} \left[ x_n \cos \theta \cos \phi + y_n \cos \theta \sin \phi + z_n \sin \theta \right]}$$
Uniform linear array (ULA)

Steering vector

\[ a(\theta) = \begin{bmatrix} 1 & e^{i2\pi f_s} & \ldots & e^{i2\pi(N-1)f_s} \end{bmatrix}^T ; \quad f_s = f_c \frac{d \sin \theta}{c} = \frac{d}{\lambda} \sin \theta \]

Shannon spatial sampling theorem

\[ |f_s| \leq 0.5 \Rightarrow d \leq \frac{\lambda}{2} \]
Covariance matrix

**Definition**

The **covariance matrix** is defined as

\[
R = \mathcal{E} \{ y(k)y^H(k) \} = \mathcal{E} \left\{ \begin{bmatrix}
y_1(k) \\
y_2(k) \\
\vdots \\
y_N(k)
\end{bmatrix} \begin{bmatrix}
y_1^*(k) \\
y_2^*(k) \\
\vdots \\
y_N^*(k)
\end{bmatrix} \right\}
\]

**Interpretation**

The \((n, \ell)\) entry \(R(n, \ell) = \mathcal{E} \{ y_n(k)y_{\ell}^*(k) \}\) measures the correlation between signals received at sensors \(n\) and \(\ell\), *at the same time index* \(k\).
Structure of the covariance matrix

Signals covariance matrix

The covariance matrix of the signal component is

\[
R = \mathcal{E}\left\{A(\theta)s(k)s^H(k)A^H(\theta)\right\} = A(\theta)R_s A^H(\theta)
\]

\[
= \sum_{p=1}^P P_p a(\theta_p) a^H(\theta_p) \quad \text{(uncorrelated signals)}
\]

Provided that \(R_s\) is full-rank (non coherent signals), the signal covariance matrix has rank \(P\) and its range space is spanned by the steering vectors \(a(\theta_p), \ p = 1, \cdots, P\).

Noise covariance matrix

Assuming spatially white noise (i.e., uncorrelated between channels) with same power on each channel, \(\mathcal{E}\left\{n(k)n^H(k)\right\} = \sigma^2 I\).
Model limitations

\( y(k) = a(\theta)s(k) + n(k) \) is an idealized model of the signals received on the array. It does not account for:

- a possibly non homogeneous propagation medium which results in coherence loss and wavefront distortions. This leads to amplitude and phase variations along the array, i.e.
  \[ y_n(k) = g_n(k) e^{i\phi_n(k)} a_n(\theta) s(k) + n_n(k). \]

- uncalibrated arrays, i.e., different amplitude and phase responses for each channel.

- wideband signals for which a time delay does not amount to a simple phase shift. In the frequency domain, one has
  \[ y(f) = a_f(\theta) s(f) + n(f) \]
  \[ a_f(\theta) = \begin{bmatrix} 1 & e^{-i2\pi f\tau(\theta)} & \ldots & e^{-i2\pi (N-1) f\tau(\theta)} \end{bmatrix}^T. \]

- possibly colored reception noise, i.e. \( \mathcal{E}\{n(k)n^H(k)\} \neq \sigma^2 I. \)
Contents

1 Introduction

2 Array processing model

3 Beamforming
   Principle
   Array beampattern
   Spatial filtering
   SNR improvement
   Adaptive beamforming
   Robust adaptive beamforming
   Partially adaptive beamforming

4 Source localization
Spatial filtering

Principle: use a **linear combination of the sensors outputs** in order to point towards a looked direction.

\[
y_F(k) = \sum_{n=1}^{N} w_n^* y_n(k) \approx a_s(k)
\]
Array beampattern

- For any weight vector \( w \), the corresponding array beampattern is defined as \( G_w(\theta) = |g_w(\theta)|^2 \) with \( g_w(\theta) = w^H a(\theta) \).
- For a uniform linear array, the natural beampattern, obtained as a simple sum (\( w_n = 1 \)) of the sensors outputs, is given by

\[
G(\theta) = |g(\theta)|^2 = \left| \frac{\sin \left[ \pi N \frac{d}{\lambda} \sin \theta \right]}{\sin \left[ \pi \frac{d}{\lambda} \sin \theta \right]} \right|^2
\]

\[
g(\theta) = \sum_{n=0}^{N-1} e^{i2\pi n \frac{d}{\lambda} \sin \theta}
\]

\[
= e^{i\pi(N-1) \frac{d}{\lambda} \sin \theta} \times \frac{\sin \left[ \pi N \frac{d}{\lambda} \sin \theta \right]}{\sin \left[ \pi \frac{d}{\lambda} \sin \theta \right]}
\]
Beampattern of the uniform linear array

\[ \theta_{3\text{dB}} \approx \frac{0.9\lambda}{N \cdot d} \]
ULa beampattern with windowing

- Rect
- Cheb 30dB
- Cheb 50dB

O. Besson (ISAE-SUPAERO)
Objective

We aim at pointing towards a given direction in order to enhance reception of the signals impinging from this direction, and to possibly mitigate interference located at other directions.

Principle

Each sensor output is weighted by $w_n^*$ before summation:

$$y_F(k) = \sum_{n=1}^{N} w_n^* y_n(k) = [w_1^* \ w_2^* \ \cdots \ w_N^*] \ y(k) = w^H y(k).$$

Question

How to choose $w$ such that, if $y(k) = a(\theta_s)s(k) + \cdots$ then at the output $y_F(k) \simeq \alpha s(k)$?
Conventional beamforming

Conventional beamforming: \( \mathbf{w} \propto \mathbf{a}(\theta_s) \)

\[
y_F(k) = \mathbf{a}^H(\theta_s)\mathbf{a}(\theta_s)s(k) \quad [\mathbf{w} = \mathbf{a}(\theta_s), \text{1 source at } \theta_s]
\]

\[
= \sum_{n=0}^{N-1} e^{-i2\pi \frac{d}{\lambda} n \sin \theta_s} \times e^{+i2\pi \frac{d}{\lambda} n \sin \theta_s} s(k)
\]

\[
= \sum_{n=0}^{N-1} s(k) = N s(k)
\]

so that the gain towards \( \theta_s \) is \textbf{maximal} and equal to \( N \). The beamformer \( \mathbf{w}_{\text{CBF}} = \mathbf{a}(\theta_s)/[\mathbf{a}^H(\theta_s)\mathbf{a}(\theta_s)] \) is referred to as the \textit{conventional beamformer}.

Principle

One compensates for the phase shift induced by propagation from direction \( \theta_s \) and then sum \textbf{coherently}. 
Array beampattern with conventional beamforming

Conventional beamformer

- $\theta_s = 0^\circ$
- $\theta_s = 30^\circ$
- $\theta_s = -45^\circ$
Before beamforming

\[ y(k) = a_s s(k) + n(k); \quad \text{SNR}_{\text{elem}} \triangleq \frac{\mathcal{E}\left\{ |s(k)|^2 \right\}}{\mathcal{E}\left\{ |n(k)|^2 \right\}} = \frac{P}{\sigma^2}. \]

After beamforming

\[ y_F(k) = w^H y(k) = w^H a_s s(k) + w^H n(k) \]

\[ \text{SNR}_{\text{array}} = \frac{|w^H a_s|^2}{\|w\|^2} \text{SNR}_{\text{elem}} \leq \|a_s\|^2 \text{SNR}_{\text{elem}} = N \times \text{SNR}_{\text{elem}} \]

with equality if \( w \propto a_s \).

White noise array gain

For any \( w \) such that \( w^H a_s = 1 \), the white noise array gain is

\[ A_{\text{WN}} = \frac{\text{SNR}_{\text{array}}}{\text{SNR}_{\text{elem}}} = \|w\|^{-2} \leq N. \]
Conventional beamforming versus adaptive beamforming

Conventional beamforming

The conventional beamformer is optimal in white noise: it amounts to minimize $\mathbf{w}^H \mathbf{w}$ (the output power in white noise) under the constraint $\mathbf{w}^H \mathbf{a}(\theta_s) = 1$. Any other direction is deemed to be equivalent ⇒ it does not take into account other signals (interference) present in some directions.

Adaptive beamforming

Adaptive beamforming takes into account these other signals. It consists in minimizing the output power $\mathcal{E} \left\{ |\mathbf{w}^H \mathbf{y}(k)|^2 \right\}$ while maintaining a unit gain towards looked direction ⇒ tends to place nulls towards interfering signals.
The received (input) signal in the presence of interference and noise is given by

\[ y(k) = a_s s(k) + y_I(k) + n(k) \]

where \( a_s \) is the actual SOI steering vector.

At the output of the beamformer

\[ y(k) = a_s s(k) + y_I(k) + n(k) \]

\[ w^H a_s s(k) + w^H [y_I(k) + n(k)] \]

input signal interference+noise
Signal to interference plus noise ratio (SINR)

**Definition of SINR**

For a given beamformer \( \mathbf{w} \), the usual figure of merit is the **signal to interference plus noise ratio (SINR)**, defined as

\[
\text{SINR}(\mathbf{w}) = \frac{\mathcal{E} \left\{ | \mathbf{w}^H \mathbf{a}_s s(k) |^2 \right\}}{\mathcal{E} \left\{ | \mathbf{w}^H [\mathbf{y}_I(k) + \mathbf{n}(k)] |^2 \right\}} = \frac{P_s |\mathbf{w}^H \mathbf{a}_s|^2}{\mathbf{w}^H \mathbf{C} \mathbf{w}}
\]

where \( \mathbf{C} = \mathcal{E} \left\{ [\mathbf{y}_I(k) + \mathbf{n}(k)] [\mathbf{y}_I(k) + \mathbf{n}(k)]^H \right\} \) stands for the **interference plus noise covariance matrix**.
Optimal beamformer: SINR maximization

Optimal beamformer

Maximize SINR while ensuring a unit gain towards $a_s$:

$$\min_w w^H C w \quad \text{subject to} \quad w^H a_s = 1$$

(optimal)

$$w_{opt} = \frac{C^{-1} a_s}{a_s^H C^{-1} a_s} \rightarrow SINR_{opt} = P_s a_s^H C^{-1} a_s$$

Remarks

- Principle is to minimize output power (when input $= y_I + n$) under the constraint that the actual steering vector $a_s$ goes non distorted.
- Neither $a_s$ nor $C$ will be known in practice: the actual steering vector may be different from its expected value and $C$ needs to be estimated from data (which contains $y_I + n$).
Minimum Variance Distortionless Response (MVDR)

**Principle**

Minimize output power \((\text{when input } = y_I + n)\) under the constraint that the **assumed** steering vector goes non distorted.

**Minimization problem and solution**

\[
\begin{aligned}
\min_w & \quad w^H C w \\
\text{subject to} & \quad w^H a_0 = 1
\end{aligned}
\] (MVDR)

where \(a_0\) is the **assumed** steering vector of the Signal of Interest (SoI). The solution is given by

\[
w_{\text{MVDR}} = \frac{C^{-1} a_0}{a_0^H C^{-1} a_0}
\]
Minimum Power Distortionless Response (MPDR)

 Principle

Minimize output power \((\text{when input} = a_s s + y_I + n)\) under the constraint that the assumed steering vector goes non distorted:

\[
\min_w w^H R w \quad \text{subject to} \quad w^H a_0 = 1
\]  

(MPDR)

where \(R(= C + P_s a_s a_s^H)\) stands for the signal plus interference plus noise covariance matrix.

Solution

\[
w_{\text{MPDR}} = \frac{R^{-1} a_0}{a_0^H R^{-1} a_0}
\]
Summary of adaptive beamformers (known covariance matrices)

<table>
<thead>
<tr>
<th>Beamformer</th>
<th>Principle</th>
<th>Weight vector</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Optimal</strong></td>
<td>( \min_w w^H C w ) s.t. ( w^H a_s = 1 )</td>
<td>( w_{opt} = \frac{C^{-1} a_s}{a_s^H C^{-1} a_s} )</td>
</tr>
<tr>
<td><strong>MVDR</strong></td>
<td>( \min_w w^H C w ) s.t. ( w^H a_0 = 1 )</td>
<td>( w_{MVDR} = \frac{C^{-1} a_0}{a_0^H C^{-1} a_0} )</td>
</tr>
<tr>
<td><strong>MPDR</strong></td>
<td>( \min_w w^H R w ) s.t. ( w^H a_0 = 1 )</td>
<td>( w_{MPDR} = \frac{R^{-1} a_0}{a_0^H R^{-1} a_0} )</td>
</tr>
</tbody>
</table>

- \( a_s (a_0) \) the actual (assumed) steering vector
- \( C = \text{cov}(y_I + n) \) and \( R = \text{cov}(a_s s + y_I + n) \)
Derivation of SINR

In the case

\[ y(k) = a_s s(k) + a_j s_j(k) + n(k) \quad [C = P_j a_j a_j^H + \sigma^2 I] \]

with INR = \( \frac{P_j}{\sigma^2} \gg 1 \), it can be shown that

\[ \text{SINR}_{\text{CBF}} \simeq \frac{P_s}{\sigma^2} \times \frac{1}{g \times \text{INR}}; \quad \text{SINR}_{\text{opt}} \simeq \frac{P_s}{\sigma^2} \times N(1 - g) \]

with \( g = \cos^2 (a_s, a_j) = |a_s^H a_j|^2 / (a_s^H a_s)(a_j^H a_j) \).

Remarks

• With CBF, the SINR decreases when \( P_j \) increases while it is independent of \( P_j \) with adaptive beamforming.
• The SINR decreases when \( a_j \to a_s \) (\( g \to 1 \)).
CBF and MVDR beampatterns

Comparison CBF–MVDR

CBF

MVDR

Angle of arrival

dB

O. Besson (ISAE-SUPAERO)

Introduction to array processing
The GSC structure can be represented as

\[
\begin{align*}
\begin{bmatrix}
    y(k) \\
    i(k) \\
    n(k)
\end{bmatrix}
&= \begin{bmatrix}
    a_0 s(k) \\
    i_1(k) \\
    n_1(k)
\end{bmatrix}
+ \begin{bmatrix}
    d(k) \\
    s(k) \\
    i_2(k) \\
    n_2(k)
\end{bmatrix}
+ \begin{bmatrix}
    B \\
    z(k)
\end{bmatrix} w_a
\end{align*}
\]

where the \((N - 1)\) columns of \(B\) form a basis of the subspace orthogonal to \(a_0\), i.e., \(B^H a_0 = 0\).

The \((N - 1)\) auxiliary channels \(z(k)\) are free of signal and enable one to infer the part of interference that went through the CBF.

\(w_a\) enables one to estimate, from \(z(k)\), the part of interference \(i_1(k)\) contained in \(d(k)\) since \(i_1(k)\) is correlated with \(z(k)\) through \(i_2(k)\).
• The GSC structure decomposes \( w \) into a component along \( a_0 \) and a component orthogonal to \( a_0 \), i.e., \( w = \alpha a_0 - w_\perp \):

\[
\begin{align*}
\begin{bmatrix} a_0 \\ a_0^H a_0 \end{bmatrix} & \begin{bmatrix} s(k) \\ i(k) \\ n(k) \end{bmatrix} \\
B & \begin{bmatrix} i_1(k) \\ n_1(k) \\ z(k) \end{bmatrix} \\
N|N-1 & N-1|1
\end{align*}
\]

\[
d(k) - w_\perp^H z(k)
\]

• The component along \( a_0 \) ensures that the constraint is fulfilled since

\[
w^H a_0 = \alpha^* a_0^H a_0 - w_\perp^H a_0 = \alpha^* a_0^H a_0 + 0 \Rightarrow \alpha = (a_0^H a_0)^{-1}
\]

• The orthogonal component \( w_\perp = Bw_a \) is chosen to minimize output power, in an unconstrained way.
• Minimization of the output power can be achieved by solving one of the two following equivalent problems:

\[
\min_{w} \quad w^H C w \\
\text{subject to} \quad w^H a_0 = 1
\]

direct form, constrained

\[
\min_{w_a} \left( w_{CBF} - B w_a \right)^H C \left( w_{CBF} - B w_a \right)
\]

generalized sidelobe canceler (GSC) form, unconstrained

• The MVDR beamformer in its GSC form is given by

\[
w_{GSC} = w_{CBF} - B w_a^* \]

where \( w_a^* \) solves the above minimization problem.
Derivation of $w_a^*$

- The power at the output of the beamformer is given by

$$
\mathcal{E} \left\{ \left| d(k) - w_a^H z(k) \right|^2 \right\} = \mathcal{E} \left\{ \left| d(k) \right|^2 \right\} - w_a^H r_{dz} - r_{dz}^H w_a + w_a^H R_z w_a
$$

$$
= \left[ w_a - R_z^{-1} r_{dz} \right]^H R_z \left[ w_a - R_z^{-1} r_{dz} \right] + \mathcal{E} \left\{ \left| d(k) \right|^2 \right\} - r_{dz}^H R_z^{-1} r_{dz}
$$

with $r_{dz} = \mathcal{E} \left\{ z(k)d^*(k) \right\}$ and $R_z = \mathcal{E} \left\{ z(k)z(k)^H \right\}$.

- The weight vector which minimizes output power is thus

$$
w_a^* = R_z^{-1} r_{dz}$$
The GSC form of the weight vector is given by

\[
\mathbf{w}_{\text{GSC}} = \mathbf{w}_{\text{CBF}} - \mathbf{BR}^{-1}_{z} \mathbf{r}_{dz}
\]

\[
= \mathbf{w}_{\text{CBF}} - \mathbf{B} \left( \mathbf{B}^H \mathbf{R}_{y} \mathbf{B} \right)^{-1} \mathbf{B}^H \mathbf{R}_{y} \mathbf{w}_{\text{CBF}}
\]

(GSC)

where \( \mathbf{R}_{y} = \mathbf{R} \) in a MPDR scenario and \( \mathbf{R}_{y} = \mathbf{C} \) in a MVDR scenario.

Since they solve the same problem \( \mathbf{w}_{\text{GSC}} = \left( \mathbf{a}_0^H \mathbf{R}_{y}^{-1} \mathbf{a}_0 \right)^{-1} \mathbf{R}_{y}^{-1} \mathbf{a}_0 \).

The SINR is inversely proportional to the output power when \( \mathbf{R}_{y} = \mathbf{C} \), i.e.,

\[
\text{SINR}_{\text{GSC}} = P_s \left[ \mathbf{w}_{\text{CBF}}^H \mathbf{C} \mathbf{w}_{\text{CBF}} - \mathbf{r}_{dz}^H \mathbf{R}_{z}^{-1} \mathbf{r}_{dz} \right]^{-1}
\]
Assume we have a reference signal $s(k)$ (e.g. pilot signal). Then, one may try to minimize the mean-square error:

$$
\mathcal{E} \left\{ |w^H y(k) - s(k)|^2 \right\} = w^H R_y w - w^H r_{ys} - r_{ys}^H w + P_s
$$

where $r_{ys} = \mathcal{E} \{ y(k) s^*(k) \}$.

The solution is given by

$$
w = R_y^{-1} r_{ys}
$$

If $r_{ys} = P_s a_s$ then $w = P_s R^{-1} a_s$, which is exactly the MPDR beamformer (without requiring knowledge of $a_s$).
Interpretation of optimal beamformer

- Assuming $J$ interfering signals, then

$$
C = \sum_{j=1}^{J} P_j a_j a_j^H + \sigma^2 I = \sum_{n=1}^{J} (\lambda_n + \sigma^2) u_n u_n^H + \sigma^2 \sum_{n=J+1}^{N} u_n u_n^H
$$

where $\mathcal{R} \{ u_1, \cdots, u_J \} = \mathcal{R} \{ a_1, \cdots, a_J \}$, i.e., principal eigenvectors span the same subspace as interference steering vectors.

- The MVDR beamformer can be rewritten as

$$
\mathbf{w}_{\text{opt}} = \alpha \left[ \mathbf{w}_{\text{CBF}} - \sum_{n=1}^{J} \frac{\lambda_n}{\lambda_n + \sigma^2} \left[ u_n^H \mathbf{w}_{\text{CBF}} \right] u_n \right]
$$

where $\alpha = (a_s^H a_s)/(a_s^H C^{-1} a_s)/\sigma^2$.
The optimal beamformer amounts to subtract from the CBF a linear combination of the $J$ principal eigenvectors of $C$.

These “eigenbeams” enable one to evaluate the part of interference that went through the conventional beamformer.
Beampatterns (CBF and eigenvectors)

- CBF
- 1st eigen-beam
- 2nd eigen-beam
- MVDR
Beampatterns (CBF and eigenvectors)
SINR versus number of eigenvectors

Signal to interference and noise ratio

Number of eigen-beams vs. dB Signal to interference and noise ratio

O. Besson (ISAE-SUPAERO)
Introduction to array processing
The optimal, MVDR and MPDR beamformers are equivalent if and only if

\[
\min_w w^H (C + P_s a_s a_s^H) w \text{ subject to } w^H a_0 = 1
\]

\(\equiv\)

\[
\min_w w^H C w \text{ subject to } w^H a_0 = 1
\]

\(\equiv\)

\[
\min_w w^H C w \text{ subject to } w^H a_s = 1
\]

which is true only when the 2 following conditions are satisfied:

1. the assumed steering vector \(a_0\) coincides with the actual steering vector \(a_s\): in practice, uncalibrated arrays or a pointing error lead to \(a_0 \neq a_s\);
2. the covariance matrix \(R\) is known: in practice, one needs to estimate it which results in estimation errors \(\hat{R} - R\).

\(\Rightarrow\) It ensues that degradation compared to \(\text{SINR}_{\text{opt}}\) is unavoidable in practice, and it can be quite different between MPDR and MVDR.
We assume that the SoI steering vector is \( \mathbf{a}_0 \) while it is actually \( \mathbf{a}_s \).

The SINR obtained with \( \mathbf{w}_{MVDR} = (\mathbf{a}_0^H \mathbf{C}^{-1} \mathbf{a}_0)^{-1} \mathbf{C}^{-1} \mathbf{a}_0 \) becomes

\[
\text{SINR}_{MVDR} = \frac{P_s |\mathbf{w}_{MVDR}^H \mathbf{a}_s|^2}{\mathbf{w}_{MVDR}^H \mathbf{C} \mathbf{w}_{MVDR}} = P_s \frac{|\mathbf{a}_0^H \mathbf{C}^{-1} \mathbf{a}_s|^2}{\mathbf{a}_0^H \mathbf{C}^{-1} \mathbf{a}_0}
\]

\[
= \text{SINR}_{opt} \times \frac{|\mathbf{a}_0^H \mathbf{C}^{-1} \mathbf{a}_s|^2}{(\mathbf{a}_0^H \mathbf{C}^{-1} \mathbf{a}_0)(\mathbf{a}_s^H \mathbf{C}^{-1} \mathbf{a}_s)}
\]

\[
= \text{SINR}_{opt} \times \cos^2 (\mathbf{a}_s, \mathbf{a}_0; \mathbf{C}^{-1}) \leq \text{SINR}_{opt}
\]
Influence of a steering vector error (MPDR)

- The MPDR beamformer can be written as

\[ w_{\text{MPDR}} = \frac{R^{-1}a_0}{a_0^H R^{-1}a_0}; \quad R = P_s a_s a_s^H + C \]

- Its SINR is decreased compared to that of the MVDR, viz

\[ \text{SINR}_{\text{MPDR}} = \frac{\text{SINR}_{\text{MVDR}}}{1 + \left(2\text{SINR}_{\text{opt}} + \text{SINR}_{\text{opt}}^2\right) \sin^2 (a_s, a_0; C^{-1})} \leq \text{SINR}_{\text{MVDR}}. \]

- The degradation is more important as \( P_s \) increases.
Influence of a steering vector error on beampatterns

\[ \theta_0 - \theta_s = 2^\circ \]

Beampatterns with pointing errors

- opt
- MVDR
- MPDR

Angle of arrival

dB

O. Besson (ISAE-SUPAERO)
Influence of a steering vector error on SINR and WNAG

![Graph showing SINR loss and white noise array gain with pointing error as a fraction of \( \theta_{3dB} \).]

- SINR loss when \( \mathbf{a}_0 \neq \mathbf{a}_s \)

- White noise array gain when \( \mathbf{a}_0 \neq \mathbf{a}_s \)

- The graphs illustrate the impact of steering vector errors on SINR and WNAG.
Let us consider an uncalibrated array with actual steering vector

\[ \tilde{a}_n(\theta) = (1 + g_n) e^{i\phi_n} a_n(\theta) \]

where \( \{g_n\} \) and \( \{\phi_n\} \) are independent random gains and phases.

For any beamformer \( \mathbf{w} \), the average value of the resulting beampattern \( \tilde{G}_w(\theta) = |\mathbf{w}^H \tilde{a}(\theta)|^2 \) is related to the nominal beampattern \( G_w(\theta) = |\mathbf{w}^H a(\theta)|^2 \) through

\[
\mathcal{E} \left\{ \tilde{G}_w(\theta) \right\} = |\gamma|^2 G_w(\theta) + \left[ 1 + \sigma_g^2 - |\gamma|^2 \right] \|\mathbf{w}\|^2
\]

where \( \sigma_g^2 = \mathcal{E} \{ |g_n|^2 \} \) and \( \gamma = \mathcal{E} \{ e^{i\phi_n} \} \).

The term proportional to \( \|\mathbf{w}\|^2 \) leads to sidelobe level increase \( \Rightarrow \) better to have high white noise array gain (low \( \|\mathbf{w}\|^2 \)).
Influence of a finite number of snapshots

- In practice, $K$ snapshots are available:

$$y(k) = a_s s(k) + y_I(k) + n(k); \quad k = 1, \ldots, K$$

- The covariance matrices are thus estimated and subsequently one can compute the corresponding beamformers as

$$\hat{R} = \frac{1}{K} \sum_{k=1}^{K} y(k)y^H(k) \quad \rightarrow w_{\text{MPDR}}^{\text{smi}} = \frac{\hat{R}^{-1}a_0}{a_0^H \hat{R}^{-1}a_0}$$

$$\hat{C} = \frac{1}{K} \sum_{k=1}^{K} y_{i+n}(k)y_{i+n}^H(k) \quad \rightarrow w_{\text{MVDR}}^{\text{smi}} = \frac{\hat{C}^{-1}a_0}{a_0^H \hat{C}^{-1}a_0}$$

where $\text{smi}$ stands for “sample matrix inversion”.

Influence of a finite number of snapshots

- The sample beamformers $w_{\text{M-DR}}^{\text{smi}}$ will differ from their ensemble counterparts $w_{\text{M-DR}}$ since $\hat{R} = R + \Delta R$ and $\hat{C} = C + \Delta C$.

- The weight vectors $w_{\text{M-DR}}^{\text{smi}}$ are random and so are their corresponding signal to noise ratios

$$\text{SINR} (w_{\text{MPDR}}^{\text{smi}}) = P_s \frac{|a_0^H \hat{R}^{-1} a_s|^2}{a_0^H \hat{R}^{-1} C \hat{R}^{-1} a_0}$$

$$\text{SINR} (w_{\text{MVDR}}^{\text{smi}}) = P_s \frac{|a_0^H \hat{C}^{-1} a_s|^2}{a_0^H \hat{C}^{-1} C \hat{C}^{-1} a_0}$$

- Important issue is speed of convergence, i.e., how large should $K$ be for $\text{SINR} (w_{\text{MPDR}}^{\text{smi}})$ or $\text{SINR} (w_{\text{MVDR}}^{\text{smi}})$ to be “close” to $\text{SINR}_{\text{opt}}$?
• When $a_0 = a_s$, the **SINR loss** $\rho_{\text{MVDR}} \in [0, 1]$

$$\rho_{\text{MVDR}} = \frac{\text{SINR}(w_{\text{MVDR}})}{\text{SINR}(w_{\text{opt}})} \overset{d}{=} \left[ 1 + \frac{\chi^2_{2(N-1)}(0)}{\chi^2_{2(K-N+2)}(0)} \right]^{-1}$$

follows a **beta distribution**, i.e.,

$$p(\rho_{\text{MVDR}}) = \frac{\Gamma(K + 1)}{\Gamma(K - N + 2)\Gamma(N - 1)} \rho_{\text{MVDR}}^{K-N+1}(1 - \rho_{\text{MVDR}})^{N-2}$$

• The expected value is $\mathcal{E}\{\rho_{\text{MVDR}}\} = (K - N + 2)/(K + 1)$, so that $\text{SINR}(w_{\text{MVDR}})$ is (on average) within 3dB of the optimal SINR for $K_{\text{MVDR}} = 2N - 3$. 
SINR loss with finite number of snapshots (MPDR)

- As for $\rho_{\text{MPDR}}$ one has

$$\rho_{\text{MPDR}} = \frac{\text{SINR}(\mathbf{w}_{\text{MPDR}}^\text{smi})}{\text{SINR}(\mathbf{w}_{\text{opt}})} = d \left[ 1 + (1 + \text{SINR}_{\text{opt}}) \frac{\chi^2_{2(N-1)}(0)}{\chi^2_{2(K-N+2)}(0)} \right]^{-1}$$

- The distribution of $\rho_{\text{MPDR}}$ is

$$p(\rho_{\text{MPDR}}) = \frac{\Gamma(K + 1)(1 + \text{SINR}_{\text{opt}})^{K-N+2}}{\Gamma(N-1)\Gamma(K-N+2)} \frac{\rho_{\text{MPDR}}^{K-N+1} (1 - \rho_{\text{MPDR}})^{N-2}}{(1 + \rho_{\text{MPDR}} \text{SINR}_{\text{opt}})^{K+1}}$$

- The average number of snapshots to achieve the optimal SINR within 3dB is about

$$K_{\text{MPDR}} \simeq (N - 1) [1 + \text{SINR}_{\text{opt}}]$$

where $\text{SINR}_{\text{opt}} \simeq N \left( \frac{P_s}{\sigma^2} \right)$. In general, $K_{\text{MPDR}} \gg K_{\text{MVDR}}$. 
Beampatterns with finite number of snapshots

![Beampatterns CBF–MPDR–MVDR](image)

Beampatterns CBF–MPDR–MVDR

- MVDR
- MPDR
- CBF

N=10, K=20

Angle of arrival

dB

N=10, K=20

O. Besson (ISAE-SUPAERO)

Introduction to array processing
Distribution of SINR loss

\[ N = 10, \quad K = 20 \]

- MVDR
- MPDR \( SINR_{opt} = -10\text{dB} \)
- MPDR \( SINR_{opt} = 0\text{dB} \)
- MPDR \( SINR_{opt} = 10\text{dB} \)
SINR versus number of snapshots

SINR versus K

- Optimum
- MVDR
- MPDR
- CBF

Number of snapshots

dB

O. Besson (ISAE-SUPAERO)
Introduction to array processing
How to make MPDR more robust?

Observations

- Estimation of covariance matrices leads to a significant SINR loss (especially for the MPDR beamformer) due to
  - the interference being less eliminated
  - a sidelobe level increase which results in a lower white noise gain.

- In case of uncalibrated arrays, steering vector errors are all the more emphasized that the white noise gain is low (or $||w||^2$ large).
How to make MPDR more robust?

White noise array gain and SINR

A possible remedy

**Enforce a minimal white noise array gain** or equivalently restrain $\|w\|^2$ in order to make the MPDR beamformer more robust.
Diagonal loading

**Principle**

One tries to solve

\[
\min_w w^H \hat{R} w \text{ subject to } w^H a_0 = 1 \text{ and } \|w\|^2 = A_{WN}^{-1}(\geq N^{-1})
\]

**Solution**

The Lagrangian is given by (with \(\lambda \in \mathbb{C}\) and \(\mu \in \mathbb{R}\))

\[
L(w, \lambda, \mu) = w^H \hat{R} w + \lambda (w^H a_0 - 1) + \lambda^* (a_0^H w - 1) + \mu (w^H w - A_{WN}^{-1})
\]

\[
= \left[ w + \lambda \left( \hat{R} + \mu I \right)^{-1} a_0 \right]^H \left( \hat{R} + \mu I \right) \left[ w + \lambda \left( \hat{R} + \mu I \right)^{-1} a_0 \right] \\
- \lambda - \lambda^* - \mu A_{WN}^{-1} - |\lambda|^2 a_0^H \left( \hat{R} + \mu I \right)^{-1} a_0.
\]
**Solution**

The solution thus takes the form $\mathbf{w}_{\text{MPDR-DL}} = -\lambda \left( \hat{\mathbf{R}} + \mu \mathbf{I} \right)^{-1} \mathbf{a}_0$. Since $\mathbf{w}_{\text{MPDR-DL}}^H \mathbf{a}_0 = 1$, it follows that

$$\mathbf{w}_{\text{MPDR-DL}} = \frac{\left( \hat{\mathbf{R}} + \mu \mathbf{I} \right)^{-1} \mathbf{a}_0}{\mathbf{a}_0^H \left( \hat{\mathbf{R}} + \mu \mathbf{I} \right)^{-1} \mathbf{a}_0}$$

and $\mu$ is selected such that $\| \mathbf{w}_{\text{MPDR-DL}} \|^{-2} = A_{\text{WN}}$. 
Diagonal loading: adaptivity versus robustness

Diagonal loading

$N$

$A_w$

MPDR

CBF

$\mu$

$\mu$

MPDR

CBF

$\mu$

Diagonal loading
Many different possibilities have been proposed to set the loading level:

- set $A_{WN}$ (slightly below $N$) and compute $\mu$ from $\|w_{MPDR-DL}\|^{-2} = A_{WN}$.
- set $\mu$ directly, generally a few decibels above white noise level (see discussion next slide about beampatterns and eigenvalues).
- set $\mu$ using the theory of ridge regression, which enables one to compute $\mu$ from data.
- use that diagonal loading is the solution to the following problem

$$\max_{P,a} \hat{R} - Paa^H \text{ for } \|a - a_0\|^2 \leq \varepsilon^2$$

and compute $\mu$ from $\varepsilon$.
- set $A_{WN}$ and compute directly the diagonally loaded beamformer in GSC form without necessarily computing $\mu$.
- ...
An interpretation of diagonal loading and the choice of $\mu$

- The array beampattern with the **true** covariance matrix is given by

$$g(\theta) = \frac{\alpha}{\sigma^2} \left\{ a_0^H a(\theta) - \sum_{n=1}^{J} \frac{\lambda_n}{\lambda_n + \sigma^2} \left[ a_0^H u_n \right] u_n^H a(\theta) \right\}$$

- The array beampattern with an **estimated** covariance matrix becomes

$$g^{smi}(\theta) = \frac{\alpha}{\hat{\lambda}_{\text{min}}} \left\{ a_0^H a(\theta) - \sum_{n=1}^{N} \frac{\hat{\lambda}_n}{\hat{\lambda}_n + \hat{\lambda}_{\text{min}}} \left[ a_0^H \hat{u}_n \right] \hat{u}_n^H a(\theta) \right\}$$

- Degradation is due to $\hat{\lambda}_{J+1} \neq \hat{\lambda}_{J+2} \neq \cdots \hat{\lambda}_N = \hat{\lambda}_{\text{min}}$.

- Replacing $\hat{R}$ by $\hat{R} + \mu I$ enables one to equalize the eigenvalues, provided that $\mu \gg \sigma^2$ and $\mu < \lambda_J$. 

O. Besson (ISAE-SUPAERO)

Introduction to array processing

70 / 116
Diagonal loading: SINR versus number of snapshots

SINR versus K – Diagonal loading

Number of snapshots

dB

SINR versus K – Diagonal loading

optimum
MPDR
MPDR−DL=5dB
MPDR−DL=10dB

O. Besson (ISAE-SUPAERO)
Introduction to array processing

71 / 116
Diagonal loading: beampatterns

Beampattern of MPDR with diagonal loading

N=10, K=20
MPDR
MPDR−DL=5dB
MPDR−DL=10dB

O. Besson (ISAE-SUPAERO)
Influence of the loading level on SINR and WNAG

SINR versus diagonal loading level

White noise gain versus diagonal loading level
To mitigate pointing errors, one can resort to multiple constraints, i.e. solve the problem

\[
\begin{align*}
\min \ w^H C w & \text{ subject to } Z^H w = d \\
\end{align*}
\]

whose solution is \( w = C^{-1} Z (Z^H C^{-1} Z)^{-1} d \).

One can use a unit gain constraint around the presumed DOA or a smoothness constraint:

\[
Z = \begin{bmatrix} a(\theta_0) & a(\theta_0 + \delta_1) & \cdots & a(\theta_0 + \delta_L) \end{bmatrix} \quad d = [1 \ 1 \ \cdots \ 1]^T
\]

\[
Z = \begin{bmatrix} a(\theta_0) & \frac{\partial a(\theta)}{\partial \theta} \bigg|_{\theta_0} & \cdots & \frac{\partial^L a(\theta)}{\partial \theta^L} \bigg|_{\theta_0} \end{bmatrix} \quad d = [1 \ 0 \ \cdots \ 0]^T
\]
Partially adaptive beamforming

**Principle**

Perform beamforming in a $R$-dimensional subspace.

**Why?**

- If interference lies in a subspace, it is meaningful and maybe beneficial to proceed in a (hopefully matched) lower-dimensional subspace in order to better remove interference.
- Rewriting $\mathbf{w}_{\text{MVDR}}$ in terms of $\text{eig}(\mathbf{C})$ leads to

$$\mathbf{w}_{\text{MVDR}} \propto \mathbf{w}_{\text{CBF}} - \sum_{n=1}^{J} \frac{\lambda_n}{\lambda_n + \sigma^2} \left[ \mathbf{u}_n^H \mathbf{w}_{\text{CBF}} \right] \mathbf{u}_n$$

- The rate of convergence of the MVDR is twice the number of d.o.f of the array ($N$): using $R < N$ d.o.f may decrease computational cost and improve rate of convergence.
Partially adaptive beamforming: structure

- Direct form:

\[
\begin{align*}
\mathbf{y}(k) & \rightarrow \mathbf{T} \rightarrow \tilde{\mathbf{y}}(k) \\
N|R & \rightarrow R|1 \\
\tilde{\mathbf{w}} & \rightarrow \tilde{\mathbf{w}}^H \tilde{\mathbf{y}}(k)
\end{align*}
\]

- GSC form:

\[
\begin{align*}
\mathbf{y}(k) & \rightarrow \mathbf{w}_{\text{CBF}} \\
\mathbf{d}(k) & \rightarrow \tilde{\mathbf{w}}^H \tilde{\mathbf{z}}(k) \\
\end{align*}
\]

\[
\begin{align*}
\mathbf{w}_{\text{CBF}} & \rightarrow \mathbf{B} \\
\mathbf{z}(k) & \rightarrow \tilde{\mathbf{z}}(k) \\
N|N-1 & \rightarrow N-1|R \\
\tilde{\mathbf{z}}(k) & \rightarrow \tilde{\mathbf{w}} \\
R|1 & \rightarrow \mathbf{y}(k)
\end{align*}
\]

The (columns of) matrices \( \mathbf{T} \) and \( \mathbf{U} \) can be viewed as beams pointing towards interference (and possibly the SoI) prior to filtering (beamspace filtering).
Derivation of the partially adaptive beamformer

Direct form

- New snapshots after transformation $\tilde{y}(k) = T^H y(k)$ whose covariance matrix is $R_{\tilde{y}} = T^H R_y T$.
- Minimization of the output power

$$\min_{\tilde{w}} \tilde{w}^H R_{\tilde{y}} \tilde{w} \text{ subject to } \tilde{w}^H \tilde{a}_0 = 1 \quad \text{(PA-DF)}$$

where $\tilde{a}_0 = T^H a_0$.
- The solution is given by

$$\tilde{w} = \alpha R_{\tilde{y}}^{-1} \tilde{a}_0 \Rightarrow w_{\text{PA-DF}} = \alpha T \left( T^H R_y T \right)^{-1} T^H a_0$$
Derivation of the partially adaptive beamformer

GSC form

- New snapshots after transformation \( \tilde{z}(k) = U^H z(k) = U^H B^H y(k) \)
  whose covariance matrix is \( R_{\tilde{z}} = U^H R_z U \).
- Minimization of the output power
  \[ \min_{\tilde{w}} E \left\{ |d(k) - \tilde{w}^H \tilde{z}(k)|^2 \right\} \quad \text{(PA-GSC)} \]
- The solution is given by
  \[ \tilde{w} = R_{\tilde{z}}^{-1} r_{d\tilde{z}} = (U^H R_z U)^{-1} U^H r_{dz} \]
  \[ w_{\text{PA-GSC}} = w_{\text{CBF}} - B U R_{\tilde{z}}^{-1} r_{d\tilde{z}} \]
Selection of matrices $\mathbf{T}$ and $\mathbf{U}$

Fixed transformations

- For instance using subarrays or spatial filtering, i.e.

\[
\mathbf{T} = \begin{bmatrix}
    a(\tilde{\theta}_1) & a(\tilde{\theta}_2) & \cdots & a(\tilde{\theta}_R)
\end{bmatrix},
\]

\[
\mathbf{U} = \mathbf{B}^H \begin{bmatrix}
    a(\tilde{\theta}_1) & a(\tilde{\theta}_2) & \cdots & a(\tilde{\theta}_R)
\end{bmatrix}
\]

- In this case, the columns of $\mathbf{U}$ can be viewed as beamformers aimed at intercepting the interference.

- Require some prior knowledge about the interference DOA in order for them to pass through the beams.
Selection of matrices $T$ and $U$

Random transformations

- The idea\(^a\) is to use $L$ matrices $U_\ell$ drawn from a uniform distribution on the manifold of semi-unitary $(N-1) \times R$ matrices, i.e.

$$U_\ell = X_\ell \left( X_\ell^H X_\ell \right)^{-H/2}; \quad X_\ell \overset{d}{=} \mathcal{CN} \left( 0, I_{N-1}, I_R \right)$$

and to average the corresponding weight vectors $\tilde{w}_\ell$, yielding

$$w = w_{CBF} - B \left[ \frac{1}{L} \sum_{\ell=1}^{L} U_\ell \left( U_\ell^H R_z U_\ell \right)^{-1} U_\ell^H r_{dz} \right]$$

$$= w_{CBF} - B \left[ \frac{1}{L} \sum_{\ell=1}^{L} X_\ell \left( X_\ell^H R_z X_\ell \right)^{-1} X_\ell^H r_{dz} \right]$$

The matrices $U_\ell$ are drawn from a uniform distribution on the manifold of semi-unitary matrices, or from a Gaussian distribution $\mathbb{CN}(0, I_{N-1}, I_R)$. 
Adaptive transformations

Matrices $T$ or $U$ depend on the snapshots. For example, in GSC form, if

$$R_z = \sum_{n=1}^{N-1} \lambda_n u_n u_n^H; \quad \lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_{N-1}$$

one can choose

- the $R$ principal eigenvectors of $R_z$ (Principal Component), i.e.
  $$U = [u_1 \cdots u_R] \Rightarrow w_{pc-gsc} = w_{CBF} - B U \Lambda^{-1} U^H r_{dz}$$

  where $\Lambda = \text{diag} \{\lambda_1, \cdots, \lambda_R\}$.

- the $R$ eigenvectors which contribute most to increasing the SINR (Cross Spectral Metric).
Partially adaptive beamforming: SINR versus $K$
Partially adaptive beamforming: SINR versus $K$

---

**SINR of PC and CSM beamformers**

- **$R=J=2$**
- **optimum MVDR**
- **PC**
- **CSM**

**Number of snapshots vs. SINR of PC and CSM beamformers**

- **dB**
- **R=J=2**

---

O. Besson (ISAE-SUPAERO)
Partially adaptive beamforming: SINR versus $R$

SINR of PC and CSM beamformers

K=20

optimum
MVDR
PC
CSM

O. Besson (ISAE-SUPAERO)
Introduction to array processing
Beamforming: synthesis

- **Conventional beamforming** \( w_{\text{CBF}} = (a_0^H a_0)^{-1} a_0 \). *Optimal in white noise*, \( \theta_{3\text{dB}} = 0.9 \left( \frac{N d}{\lambda} \right)^{-1} \), sidelobes at \(-13\text{dB}\).

- **Adaptive beamforming** \( w_{\text{opt}} \propto C^{-1} a_s \), \( w_{\text{MVDR}} \propto C^{-1} a_0 \), \( w_{\text{MPDR}} \propto R^{-1} a_0 \)
  - all equivalent if \( R, C \) known and \( a_s = a_0 \)
  - \( \text{SINR}_{\text{opt}} \triangleright \text{SINR}_{\text{MVDR}} \gg \text{SINR}_{\text{MPDR}} \) when \( a_s \neq a_0 \)
  - \( \text{SINR}_{\text{MVDR-SMI}} \gg \text{SINR}_{\text{MPDR-SMI}} \): convergence for about \( 2N \) snapshots for MVDR, \( N \times \text{SINR}_{\text{opt}} \) for MPDR

- **Diagonal loading**: *helps to mitigate both finite-sample errors and steering vector errors*. Especially useful in MPDR context with low power signal of interest.

- **Partially adaptive beamforming**: enables one to achieve *faster convergence* by operating in low-dimensional subspace. Especially effective with strong, low-rank interference.
Introduction to array processing

1. Introduction

2. Array processing model

3. Beamforming

4. Source localization
   - Non parametric methods (beamforming)
   - Parametric methods for DOA estimation
The direction of arrival estimation problem

Problem formulation
Given a collection of $K$ snapshots which can possibly be modeled as $y(k) = \sum_{p=1}^{P} a(\theta_p)s_p(k) + n(k)$, estimate the directions of arrival (DoA) $\theta_1, \ldots, \theta_P$:

$y(k) \overset{?}{=} \sum_{p=1}^{P} a(\theta_p)s_p(k) + n(k) \rightarrow \hat{\theta}_1, \ldots, \hat{\theta}_P$

Approaches
- Non parametric approaches which do not necessarily rely on a model for $y(k)$: similar to Fourier-based methods in time domain;
- Parametric approaches where a model is assumed and its properties (algebraic structure, distribution) are exploited.
Beamforming for direction finding purposes

- The idea is to form a beam $w(\theta)$ for each angle $\theta$ and to evaluate the power $\mathcal{E}\left\{ |y_F(k)|^2 \right\} = \mathcal{E}\left\{ |w^H(\theta)y(k)|^2 \right\}$ at the output of the beamformer versus $\theta$:

$\begin{align*}
y(k) & \rightarrow w(\theta) \\
P_w(\theta) &= \mathcal{E}\{ |w^H(\theta)y(k)|^2 \}
\end{align*}$

- Large peaks should provide the directions of arrival:
Beamforming for direction finding purposes

CBF and Capon

The conventional beamformer as well as the MPDR beamformer can be used, which yields

\[
\mathcal{E} \left\{ |y_F(k)|^2 \right\} = \frac{\mathbf{a}^H(\theta) \mathbf{R} \mathbf{a}(\theta)}{N^2} \left[ \mathbf{w}(\theta) = \frac{\mathbf{a}(\theta)}{\mathbf{a}^H(\theta) \mathbf{a}(\theta)} \right] \quad \text{(CBF)}
\]

\[
= \frac{1}{\mathbf{a}^H(\theta) \mathbf{R}^{-1} \mathbf{a}(\theta)} \left[ \mathbf{w}(\theta) = \frac{\mathbf{R}^{-1} \mathbf{a}(\theta)}{\mathbf{a}^H(\theta) \mathbf{R}^{-1} \mathbf{a}(\theta)} \right] \quad \text{(Capon)}
\]

In practice

With \( K \) snapshots available, \( \mathbf{R} \) is estimated as

\[
\hat{\mathbf{R}} = \frac{1}{K} \sum_{k=1}^{K} \mathbf{y}(k) \mathbf{y}^H(k)
\]
The estimated power at the output of the CBF writes

\[ P_{\text{CBF}}(\theta) = \frac{1}{N^2} a^H(\theta) \hat{R} a(\theta) \]

\[ = \frac{1}{KN^2} \sum_{k=1}^{K} |a^H(\theta)y(k)|^2 \]

\[ = \frac{1}{KN^2} \sum_{k=1}^{K} \left| \sum_{n=1}^{N} y_n(k)e^{-i2\pi(n-1)f} \right|^2 \]

where \( f = \frac{d}{\lambda} \sin \theta \).

The inner sum is recognized as the (spatial) Fourier transform of each snapshot.
Comparison CBF-Capon (low resolution scenario)

\[ \theta = [-30^\circ, 10^\circ, 20^\circ] \]

\[ \theta_{3dB} \sim 5.1^\circ \]
Comparison CBF-Capon (high resolution scenario)

\[ \theta = [-30^\circ, 10^\circ, 15^\circ] \]

\[ \theta_{3dB} \sim 5.1^\circ \]
Model-based methods

Principle

Based on the model

\[ y(k) = A(\theta)s(k) + n(k) \]

where \( \theta = [\theta_1 \ \theta_2 \ \cdots \ \theta_P]^T \),

\[ A(\theta) = [a(\theta_1) \ a(\theta_2) \ \cdots \ a(\theta_P)] \]

\[ s(k) = [s_1(k) \ s_2(k) \ \cdots \ s_P(k)]^T \]

and \( a(\theta) \) stands for the steering vector.
Classes of methods

- **Maximum Likelihood methods** are based on maximizing the likelihood function, which amounts to finding the unknown parameters which make the observed data the more likely.

- **Subspace-based methods** rely on the fact that the signal subspace coincides with the subspace spanned by the principal eigenvectors of $\mathbf{R}$. Moreover, the latter is orthogonal to the subspace spanned by the minor eigenvectors. These two algebraic properties are exploited for direction finding.

- **Covariance matching** relies on a model $\mathbf{R}(\eta)$ for the covariance matrix and looks for the model parameters which minimize the distance between $\mathbf{R}(\eta)$ and the sample covariance matrix $\hat{\mathbf{R}}$. 
Maximum Likelihood Estimation

- The MLE consists in finding the parameter vector $\eta$ which maximizes the likelihood function $p(Y; \eta)$ of the snapshots $Y = [y(1) \ y(2) \ \cdots \ y(k)]$, where $\eta$ is the model parameter vector.

😊 Asymptotically efficient.

😢 Multi-dimensional optimization problem (usually) $\Rightarrow$ computational complexity, possible convergence to local maxima.
• Assume that $s(k)$ is Gaussian distributed with $\mathcal{E}\{s(k)\} = 0$, and a covariance matrix $R_s = \mathcal{E}\{s(k)s^H(k)\}$ which is full rank.

• The distribution of the snapshots is thus given by

$$y(k) \sim \mathcal{CN}\left(0, R = A(\theta)R_sA^H(\theta) + \sigma^2 I\right)$$

• The likelihood function can be written as

$$p(Y; \eta) = \prod_{k=1}^{K} \pi^{-N} |R|^{-1} e^{-y(k)^H R^{-1} y(k)}$$
The ML estimate is obtained as

\[ \hat{\eta} = \arg \min_{\theta, R_s, \sigma^2} - \log p(Y; \eta) \]

\[ = \arg \min_{\theta, R_s, \sigma^2} \log |R| + \text{Tr} \left\{ R^{-1} \hat{R} \right\} \]

Closed-form solutions for \( \sigma^2 \) and \( R_s \) can be obtained so that the likelihood function is concentrated, yielding a minimization over the angles only:

\[ \hat{\theta}^{\text{sto}} = \arg \min_{\theta} \log \left| A(\theta) \hat{R}_s(\theta) A^H(\theta) + \hat{\sigma}^2(\theta) I \right| \]
Deterministic (conditional) MLE

- The signal waveforms are assumed deterministic so that
  \[ y(k) \sim \mathcal{CN} \left( A(\theta)s(k), \sigma^2 I \right) \]

- The MLE is now given by
  \[
  \hat{\eta} = \arg \min_{\theta, s(k), \sigma^2} NK \log \sigma^2 + \sigma^{-2} \sum_{k=1}^{K} \| y(k) - A(\theta)s(k) \|^2
  \]

- The likelihood function can be concentrated with respect to all \( s(k) \) and \( \sigma^2 \), and finally
  \[
  \hat{\theta}^{\text{det}} = \arg \min_{\theta} \text{Tr} \left\{ P_A(\theta)^{\perp} \hat{R} \right\}
  \]

- For a single source \( \hat{\theta}^{\text{det}} = \arg \max_{\theta} \frac{1}{N} a^H(\theta)\hat{R}a(\theta) \equiv \text{CBF} \).
Subspace methods

Eigenvalue decomposition of the covariance matrix

If $P$ signals are present, one has

$$R = A(\theta_0)R_sA^H(\theta_0) + \sigma^2 I = \sum_{p=1}^{P} \lambda_p u_p u_p^H + \sigma^2 I$$

$$= \sum_{p=1}^{P} (\lambda_p + \sigma^2) u_p u_p^H + \sigma^2 \sum_{p=P+1}^{N} u_p u_p^H = U_s \Lambda_s U_s^H + \sigma^2 U_n U_n^H$$
Subspace methods

Signal and noise subspaces

Since

\[ RU_n = \sigma^2 U_n = A(\theta_0) R_s A^H(\theta_0) U_n + \sigma^2 U_n \]

\[ \Rightarrow A^H(\theta_0) U_n = 0 \]

we have

\[ \mathcal{N} \{ A^H(\theta_0) \} = \mathcal{R} \{ U_n \} = \mathcal{R} \{ U_s \}^\perp = \mathcal{R} \{ A(\theta_0) \}^\perp \]

\[ \Rightarrow \mathcal{R} \{ U_s \} = \mathcal{R} \{ A(\theta_0) \} \]

The signal subspace is spanned by \( U_s \): it is thus orthogonal to \( U_n \).
• The signal steering vectors are orthogonal to $U_n$

$$u_n^H a(\theta_p) = 0 \Rightarrow a^H(\theta_p) U_n U_n^H a(\theta_p) = 0$$

• **One looks for the $P$ largest maxima** of

$$V_{MUSIC}(\theta) = \frac{1}{a^H(\theta) \hat{U}_n \hat{U}_n^H a(\theta)}$$

• For a ULA, one can either compute the $P$ roots (root-MUSIC) of

$$V_{MUSIC}(z) = a^T(z^{-1}) \hat{U}_n \hat{U}_n^H a(z)$$

closest to the unit circle, where $a(z) = [1 \ z \ \ldots \ z^{N-1}]^T$.

• Many variants around MUSIC, e.g., SSMUSIC (Mc Cloud & Scharf).
Subspace Fitting

- Since $\mathcal{R}\{\hat{U}_s\} = \mathcal{R}\{A(\theta_0)\}$, there exists a full-rank matrix $T$ ($P \times P$) such that
  \[ U_s = A(\theta_0)T \]

- The idea is to look for the DOA which minimize the error between the subspaces spanned by $\hat{U}_s$ and $A(\theta)$:

  \[ \hat{\theta}, \hat{T} = \arg \min_{\theta,T} \| \hat{U}_s - A(\theta)T \|^2_W \]

  \[ = \arg \min_{\theta,T} \text{Tr} \left\{ \left[ \hat{U}_s - A(\theta)T \right] W \left[ \hat{U}_s - A(\theta)T \right]^H \right\} \]
Subspace Fitting

• There exists a closed-form solution for $T$ and finally

$$\hat{\theta}^{SSF} = \arg\min_{\theta} \text{Tr} \left\{ P_{A}(\theta) \hat{U}_s W \hat{U}_s^H \right\}$$

• Alternative: use the fact that

$$\mathcal{R} \{U_n\} = \mathcal{N} \{ A^H(\theta_0) \} \Rightarrow U_n^H A(\theta_0) = 0$$

and estimate the angles as

$$\hat{\theta}^{NSF} = \arg\min_{\theta} \left\| \hat{U}_n^H A(\theta) \right\|_W^2$$
We assume that the array is composed of 2 sub-arrays which are related to by a known displacement. Then

\[
A_2 = A_1 \Phi = A_1 \begin{pmatrix}
e^{i\omega_c \tau(\theta_1)} \\
e^{i\omega_c \tau(\theta_2)} \\
\vdots \\
e^{i\omega_c \tau(\theta_P)}
\end{pmatrix}
\]
• The signals received on the two sub-arrays can be written as

\[ y_1(k) = A_1 s(k) + n_1(k) \]
\[ y_2(k) = A_1 \Phi s(k) + n_2(k) \]

• Let

\[ z(k) = \begin{bmatrix} y_1(k) \\ y_2(k) \end{bmatrix} = \begin{bmatrix} A_1 \\ A_1 \Phi \end{bmatrix} s(k) + \begin{bmatrix} n_1(k) \\ n_2(k) \end{bmatrix} = \bar{A} s(k) + \bar{n}(k) \]

and \( R_z = \mathcal{E} \{ z(k) z^H(k) \} \) be the covariance matrix of \( z(k) \).
• If \( R_z = U_s \Lambda_s U_s^H + U_n \Lambda_n U_n^H \) then

\[
U_s = \bar{A} T \Rightarrow \begin{bmatrix} U_1 \\ U_2 \end{bmatrix} = \begin{bmatrix} A_1 \\ A_1 \Phi \end{bmatrix} T
\]
\[
\Rightarrow U_1 = A_1 T \text{ et } U_2 = A_1 \Phi T
\]
\[
\Rightarrow U_2 = U_1 T^{-1} \Phi T
\]
\[
\Rightarrow U_2 = U_1 \Psi
\]

• The eigenvalues of \( \Psi \) are \( \left\{ e^{i \omega_c \tau(\theta_p)} \right\}_{p=1}^P \).
Low-resolution scenario

Eigenvalues of the covariance matrix

- $\text{dB}$
- $\text{R theory}$
- $\text{R estimated}$

O. Besson (ISAE-SUPAERO)
Introduction to array processing
Low-resolution scenario

\[ \theta = [-30^\circ, 10^\circ, 20^\circ] \]

\[ \theta_{3dB} \sim 5.1^\circ \]

Comparison CBF–MUSIC

Angle of arrival

dB

\[ \theta = [-30^\circ, 10^\circ, 20^\circ] \]

\[ \theta_{3dB} \sim 5.1^\circ \]

CBF

MUSIC

Roots

root–MUSIC

arbitrary in \( R(U_n) \)

O. Besson (ISAE-SUPAERO)  Introduction to array processing  109 / 116
High-resolution scenario

Eigenvalues of the covariance matrix

-5
0
5
10
15
20
25
30

dB

R theory
R estimated

O. Besson (ISAE-SUPAERO)
Introduction to array processing
High-resolution scenario

Comparison CBF–MUSIC

$\theta = [-30^\circ, 10^\circ, 13^\circ]$  
$\theta_{3\text{dB}} \sim 5.1^\circ$

O. Besson (ISAE-SUPAERO)
Introduction to array processing

111 / 116
Covariance matching

- The covariance matrix is given by $R(\theta, P, \sigma) = R_s(\theta, P) + Q(\sigma)$

$$r = \text{vec}(R) = \Psi(\theta)P + \Sigma\sigma = \begin{bmatrix} \Psi(\theta) & \Sigma \end{bmatrix} \begin{bmatrix} P \\ \sigma \end{bmatrix} \triangleq \Phi(\theta)\alpha$$

- The parameters are estimated by minimizing the error between $R$ and its estimate $\hat{R}$:

$$\hat{\theta}, \hat{\alpha} = \arg \min \left[ \hat{r} - \Phi(\theta)\alpha \right] W^{-1} \left[ \hat{r} - \Phi(\theta)\alpha \right]$$

- The criterion can be concentrated with respect to $\alpha$: minimization with respect to $\theta$ only.
Covariance matching

- In case of independent Gaussian distributed snapshots, $W_{\text{opt}} = R^T \otimes R$ and covariance matching estimates are asymptotically (i.e. when $K \to \infty$) equivalent to ML estimates.
- In contrast to MLE, no need for assumptions on the pdf, only an assumption on $R$. The criterion is usually simpler to minimize.
- Covariance matching can be used with full-rank covariance matrix $R_s$ while subspace methods require the latter to be rank deficient.
<table>
<thead>
<tr>
<th>Hypotheses</th>
<th>Algorithm</th>
<th>Performance</th>
<th>Problems</th>
</tr>
</thead>
<tbody>
<tr>
<td>ML</td>
<td>distribution</td>
<td>optimization</td>
<td>optimal</td>
</tr>
<tr>
<td>COMET</td>
<td>$\mathbb{R}$</td>
<td>optimization</td>
<td>$\simeq$ optimal</td>
</tr>
<tr>
<td>MUSIC</td>
<td>$\mathbb{R}$</td>
<td>EVD</td>
<td>$\simeq$ optimal</td>
</tr>
</tbody>
</table>
Conclusions

- Array processing, thanks to additional degrees of freedom, enables one to perform spatial filtering of signals.
- Adaptive beamforming, possibly with reduced-rank transformations, enables one to achieve high SINR with a fast rate of convergence in adverse conditions (interference, noise).
- Robustness issues are of utmost importance in practical systems, and should be given a careful attention.
- Non-parametric direction finding methods are simple and robust but may suffer from a lack of resolution.
- Parametric methods offer high resolution, often at the price of degraded robustness.
References