Source Separation and Beamforming Background

Ian K. Proudler

Dept. of Electronic & Electrical Engineering, Univerity of Strathclyde

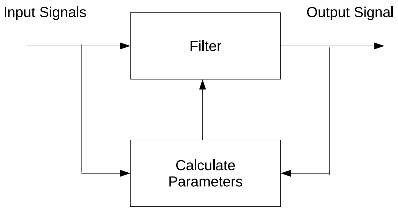
UDRC Summer School, 30th June, 2022.

Source Separation and Beamforming Background: Overview

1. Overview
2. Signal Separation
3. Non-adaptive beamforming
4. Adaptive signal processing for beamforming
5. Application of linear algebra to array problems
6. More adaptive signal processing for beamforming
7. Blind source separation
8. Summary

Signal Separation

- Signal separation requires two components:



- A parametrised mechanism to separate the signals (a “filter”)
- A means to select the parameters
- Performance limited by ‘optimum’ filter

- Conventionally we have two “filter” mechanisms:
  - Temporal filter – separate by frequency
  - Spatial filter (aka beamformer) – separate by AOA
- Could use a nonlinear filter (if you can think of one!)
- We will focus on narrowband beamforming in this talk
- Broadband beamforming requires a space-time filter

Signal Separation

- Performance limited by ‘optimum’ filter
- Narrowband beamforming $\rightarrow$ signals must have sufficiently different angles of arrival (AOA)

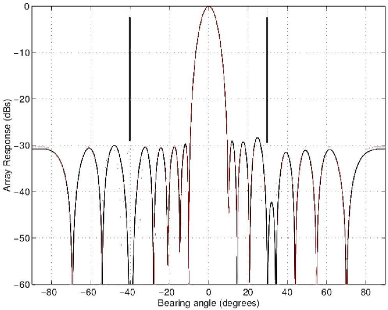
- Parameter selection – the interesting part

- Three cases:
  - Non-adaptive – we know everything about the scenario
  - “Adaptive” – we don’t know everything
  - “Blind” – we don’t know anything (sort of)

- Important parameters:
  - AOA of signals
  - Array calibration
  - Noise statistics

Non-Adaptive Source Separation

- Covered in talk by Prof. Weiss



- Beamformer weights via constrained optimisation (offline)
- Gain towards wanted signal $= 1$
- Gain towards other signals $= 0$
- Noise gain as small as possible

- Lots of good optimisation algorithms (DSP text books e.g. Rabiner & Gold - Temporal filters but basically the same for beamforming)
- Only $(N-1)$ nulls
- Spatially distributed noise can't be removed only suppressed

Adaptive Source Separation

- Usually called Adaptive Beamforming
- Assume the known parameters are:
  - AOA of the wanted signal(s)
  - Array calibration
- Beamformer weights via constrained optimisation but online this time
- Gain towards wanted signal $= 1$
- Minimise energy of output
- NB. Could use an AOA algorithm here and fixed beamforming but computationally costly

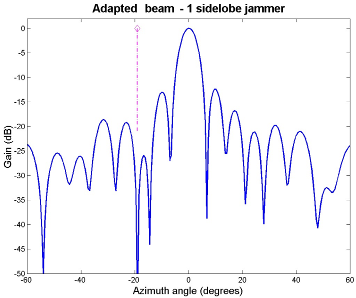
Adaptive Source Separation

- Nomenclature:
  - Steering vectors $\mathbf{A}$ relate array data to signals (see Prof. Weiss' talk)
$$\mathbf{X} = \mathbf{A}\mathbf{S}$$
  - Beamformer weights: $\mathbf{w}$
  - Sensor data at time n: $\mathbf{x}(n)$
  - Output at time n: $y(n) = \mathbf{w}^H\mathbf{x}(n)$
  - Energy in output: $J = \sum_{n=0}^{N-1} |y(n)|^2 = ||\mathbf{w}^H\mathbf{X}\mathbf{X}^H\mathbf{w}||_2^2$
  - Data matrix: $\mathbf{X} = [\mathbf{x}(0), \mathbf{x}(1), ..., \mathbf{x}(N-1)]$
  - Constraint: $\mathbf{w}^H\mathbf{a}(\theta) = 1$
  - Sample covariance matrix: $\mathbf{R} = \mathbf{X}\mathbf{X}^H$

Minimum Variance Distortionless Response (MVDR)

- Minimum Variance := Minimise energy of output
- Distortionless Response := Gain towards wanted signal = 1

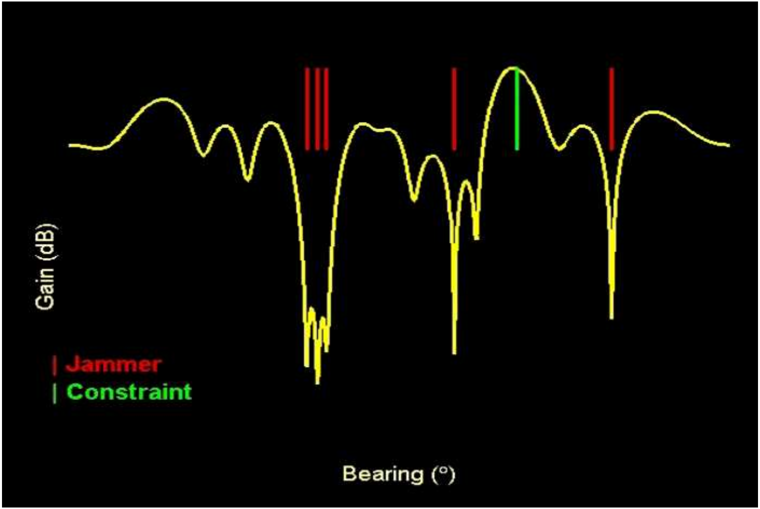
$$\mathbf{w} = \frac{\mathbf{R}^{-1}\mathbf{a}(\theta)}{\mathbf{a}^H(\theta)\mathbf{R}^{-1}\mathbf{a}(\theta)}$$



- Gain towards wanted signal = 1
- Small gain (null) towards other signal
- Noise gain not controlled
  In fact adapted to that particular noise realization

Minimum Variance Distortionless Response (MVDR)

- Multiple noise realizations (blocks of data)



Minimum Variance Distortionless Response (MVDR)

- Stabilisation procedures: there are many different ways of reducing the effects of adapting to the noise realizations.
- All effectively try to 'remove' influence of noise
- Diagonal loading

$$\mathbf{w} = \text{Arg Min}\left(||\mathbf{w}^H\left(\mathbf{R}+\mu I\right)\mathbf{w}||_2^2\right) st.\mathbf{w}^H\mathbf{a}(\theta) = 1$$

- Penalty Function Method

$$\mathbf{w} = \text{Arg Min}\left(||\mathbf{w}^H\mathbf{R}\mathbf{w}||_2^2 + \kappa||\mathbf{w}-\mathbf{w}_0||_2^2\right)$$

  "Soft" constraint makes the adapted beam pattern lie close to the desired pattern.

- "Noise" subspace manipulation: Average noise subspace eigenvalues – need some Linear Algebra

Linear Algebra

- MVDR weight vector depends on covariance matrix $\mathbf{R}$
- This matrix has structure that can be exploited
- We can use linear algebra to study / manipulate the covariance matrix

- Topics:
  - Eigenvalue decomposition of Hermitian matrix
  - Eigenvectors are not steering vectors
  - Eigenvalue spectrum
  - Signal and Noise Subspaces
  - Rotation Matrices
  - Singular Value Decomposition

Eigenvalue Decomposition

- The covariance matrix is Hermitian (symmetric)

$$\mathbf{R}^H = \left(\mathbf{X}\mathbf{X}^H\right)^H = \mathbf{X}\mathbf{X}^H = \mathbf{R}$$

- Eigenvalue decomposition of Hermitian matrix

$$\mathbf{R} = \mathbf{U}\mathbf{\Lambda}\mathbf{U}^H$$

- Eigenvectors: $\mathbf{U}$ is a unitary matrix (a rotation in n-D space)

$$\mathbf{U}^H\mathbf{U} = I$$

- Eigenvalues: $\mathbf{\Lambda}$ is diagonal, all elements are real and ≥ 0
- Rank of $\mathbf{R}$ is number of non-zero eigenvalues

Eigenvalue Decomposition

- Note the eigenvectors are not steering vectors
- Data and Covariance Matrices

$$\mathbf{X} = \mathbf{A}\mathbf{S} \qquad \mathbf{R} = \mathbf{X}\mathbf{X}^H = \mathbf{A}\mathbf{S}\mathbf{S}^H\mathbf{A}^H$$

- For independent signals $\mathbf{D} \equiv \mathbf{S}\mathbf{S}^H$ is diagonal with ≥ 0 entries.
- Consider the eigenvalue Decomposition

$$\mathbf{R} = \mathbf{U}\mathbf{\Lambda}\mathbf{U}^H = \mathbf{A}\mathbf{D}\mathbf{A}^H$$

- Tempting to assume that

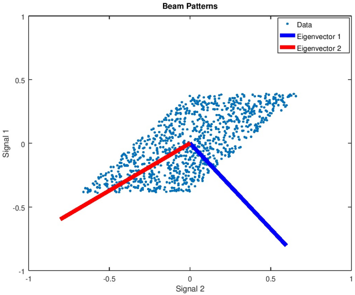
$$\mathbf{U}\mathbf{\Lambda}^{1/2} = \mathbf{A}\mathbf{D}^{1/2}$$

  which would mean that the eigenvectors are proportional to the steering vectors
- But there is an implied 'hidden' unitary matrix (SVD)

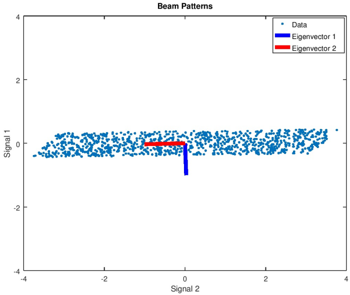
$$\mathbf{X} = \mathbf{U}\mathbf{\Lambda}^{1/2}\mathbf{V}^H \Rightarrow \mathbf{R} = \mathbf{U}\mathbf{\Lambda}^{1/2}\cancel{\mathbf{V}^H\mathbf{V}}\mathbf{\Lambda}^{1/2}\mathbf{U}^H$$

Eigenvalue Decomposition

- Eigenvectors are not steering vectors



2 equal power signals



2 signals with power ratio 10:1

- Scatter plot
- Eigenvectors of covariance matrix point in direction of maximum energy whilst being orthogonal to each other
- Eigenvectors approximately steering vectors when powers are dissimilar

Eigenvalue Decomposition

- The EVD can, however, separate ‘signals’ from ‘noise’ if SNR is high enough. Consider two signals

$$\mathbf{X} = \mathbf{a}(\theta_1)\mathbf{s}_1^T + \mathbf{a}(\theta_2)\mathbf{s}_2^T + \mathcal{N}$$

- Covariance matrix

$$\mathbf{R} = \mathbf{X}\mathbf{X}^H = \mathbf{A}\mathbf{D}\mathbf{A}^H + \sigma^2 I$$

$$\mathbf{A} = \left[\begin{array}{cc} \mathbf{a}(\theta_1) & \mathbf{a}(\theta_2) \end{array}\right] \qquad \mathbf{D} = \left[\begin{array}{cc} P_1 & 0 \\ 0 & P_2 \end{array}\right]$$

- $\mathbf{A}\mathbf{D}\mathbf{A}^H$ is rank two. EVD:

$$\mathbf{A}\mathbf{D}\mathbf{A}^H = \mathbf{U}\left[\begin{array}{cc} \mathbf{\Lambda}_\mathbf{A} & 0 \\ 0 & 0 \end{array}\right]\mathbf{U}^H$$

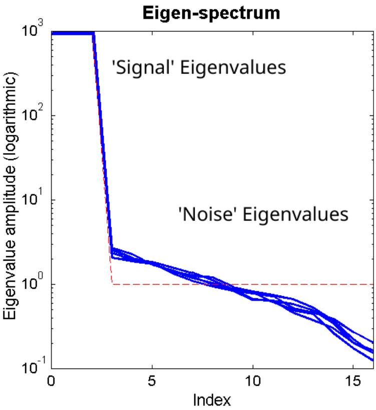
- Covariance matrix EVD (add noise)

$$\mathbf{R} = \mathbf{U}\left[\begin{array}{cc} \mathbf{\Lambda}_\mathbf{A} & 0 \\ 0 & 0 \end{array}\right]\mathbf{U}^H + \sigma^2 I = \mathbf{U}\left[\begin{array}{cc} \mathbf{\Lambda}_\mathbf{A} + \sigma^2 I & 0 \\ 0 & \sigma^2 I \end{array}\right]\mathbf{U}^H$$

Eigenvalue Spectrum

- Eigenvalue spectrum

$$\begin{bmatrix} \mathbf{\Lambda_A} + \sigma^2 \\ \sigma^2 I \end{bmatrix}$$



- Two large eigenvalues
- Five noise realizations
- Noise eigenvalues not the same and not equal what theory suggests – finite data

University of Strathclyde Engineering

Signal and Noise Subspaces

- Consider the covariance matrix EVD (replace 'theoretical' σ^2 by $\mathbf{\Lambda_N}$)

$$\mathbf{R} = \mathbf{U} \begin{bmatrix} \mathbf{\Lambda_A} + \mathbf{\Lambda_{N1}} & 0 \\ 0 & \mathbf{\Lambda_{N2}} \end{bmatrix} \mathbf{U}^H$$

- Partition eigenvectors (assuming $\mathbf{\Lambda_A} + \mathbf{\Lambda_{N1}} > \mathbf{\Lambda_{N2}}$)

$$\mathbf{U} = \begin{bmatrix} \mathbf{U_1} & \mathbf{U_2} \end{bmatrix}$$

- Orthogonal subspaces: 'Signal and Noise' and 'Noise'

$$\mathbf{U_1}^H \mathbf{U_1} = I \qquad \mathbf{U_1}^H \mathbf{U_2} = 0$$

- Then the covariance matrix EVD becomes

$$\mathbf{R} = \underbrace{\mathbf{U_1} \left(\mathbf{\Lambda_A} + \mathbf{\Lambda_{N1}} \right) \mathbf{U_1}^H}_{\text{Signal plus Noise}} + \underbrace{\mathbf{U_2} \left(\mathbf{\Lambda_{N2}} \right) \mathbf{U_2}^H}_{\text{Noise only}}$$

Signal and Noise Subspaces

- Noise reduction: The covariance matrix EVD is

$$\mathbf{R} = \underbrace{\mathbf{U_1}\left(\mathbf{\Lambda_A} + \mathbf{\Lambda_{N1}}\right)\mathbf{U_1}^H}_{\text{Signal plus Noise}} + \underbrace{\mathbf{U_2}\left(\mathbf{\Lambda_{N2}}\right)\mathbf{U_2}^H}_{\text{Noise only}}$$

- But the 'Signal plus Noise' and 'Noise only' subspaces are orthogonal

$$\mathbf{U_1}^H\mathbf{U_1} = I \qquad \mathbf{U_1}^H\mathbf{U_2} = 0$$

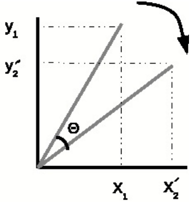
- So that

$$\begin{aligned}\tilde{\mathbf{R}} = \; & \mathbf{U_1}\mathbf{U_1}^H\mathbf{R}\overbrace{\mathbf{U_1}\mathbf{U_1}^H}^{\text{Projection Operator}} = \mathbf{U_1}\left(\mathbf{\Lambda_A} + \mathbf{\Lambda_{N1}}\right)\mathbf{U_1}^H \\ & = \mathbf{U}\begin{bmatrix} \mathbf{\Lambda_A} + \mathbf{\Lambda_{N1}} & 0 \\ 0 & 0 \end{bmatrix}\mathbf{U}^H\end{aligned}$$

- Thus $\tilde{\mathbf{R}}$ is a covariance matrix for the signals but with less noise

Rotation Matrices

- Eigenvectors: $\mathbf{U}$ is a unitary matrix $\mathbf{U}^H\mathbf{U} = I$
- Often calculated by multiple applications of Givens rotations:



$$\begin{bmatrix} \cos(\theta) & \sin(\theta) \\ -\sin(\theta)^* & \cos(\theta) \end{bmatrix} \begin{bmatrix} x_1 \\ y_1 \end{bmatrix} = \begin{bmatrix} x_2' \\ y_2' \end{bmatrix}$$

- $\mathbf{U}$ can be build up by embedding Givens rotation in N-D space

$$\mathbf{U} = [\bullet]\ldots \begin{bmatrix} I & 0 & 0 & 0 & 0 \\ 0 & \cos(\theta) & 0 & \sin(\theta) & 0 \\ 0 & 0 & I & 0 & 0 \\ 0 & -\sin(\theta)^* & 0 & \cos(\theta) & 0 \\ 0 & 0 & 0 & I & \end{bmatrix} \ldots [\bullet]$$

- Hence $\mathbf{U}$ be considered as a rotation in N-dimensional space – useful for blind signal separation theory

Singular Value Decomposition

- Not all matrices of interest are Hermitian
- Singular value decomposition of a matrix $\mathbf{X}$:

$$\mathbf{X} = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^H$$

If $\mathbf{X}$ is $N \times M$: $\mathbf{U}$ is $N \times N$, $\mathbf{\Sigma}$ is $N \times M$, and $\mathbf{V}$ is $M \times M$

- Singular vectors: $\mathbf{U}$ and $\mathbf{V}$ are unitary matrices
- Singular values: $\mathbf{\Sigma}$ is diagonal, all elements are ≥ 0
- Rank of $\mathbf{X}$ is number of non-zero singular values
- Relation to EVD

$$\mathbf{R} = \mathbf{X}\mathbf{X}^H = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^H\mathbf{V}\mathbf{\Sigma}\mathbf{U}^H = \mathbf{U}\mathbf{\Sigma}^2\mathbf{U}^H$$

Eigenvalues are the square of the singular values

- Recall we used the SVD to show that eigenvalues are not steering vectors

Stabilized MVDR Beamformer

- Back to beamforming ....
- Recall basic MVDR beamformer suffers from weight jitter
- Covariance matrix EVD

$$\mathbf{R} = \begin{bmatrix} \mathbf{U_1} & \mathbf{U_2} \end{bmatrix} \begin{bmatrix} \mathbf{\Lambda_A} + \Lambda_{N1} & 0 \\ 0 & \Lambda_{N2} \end{bmatrix} \begin{bmatrix} \mathbf{U_1}^H \\ \mathbf{U_2}^H \end{bmatrix}$$

- Idea 1: Subspace Projection: remove noise
  Orthogonal subspaces: $\mathbf{U_1}^H \mathbf{U_1} = I$, $\mathbf{U_1}^H \mathbf{U_2} = 0$

$$\hat{\mathbf{X}} = \overbrace{\mathbf{U_1} \mathbf{U_1}^H}^{\text{Projection Operator}} \mathbf{X}$$

$$\hat{\mathbf{R}} = \hat{\mathbf{X}} \hat{\mathbf{X}}^H = \begin{bmatrix} \mathbf{U_1} & 0 \end{bmatrix} \begin{bmatrix} \mathbf{\Lambda_A} + \Lambda_{N1} & 0 \\ 0 & \cancel{\Lambda_{N2}} \end{bmatrix} \begin{bmatrix} \mathbf{U_1}^H \\ 0 \end{bmatrix}$$

- Issues with rank deficient $\hat{\mathbf{R}}$ since it is not invertible

Stabilized MVDR Beamformer

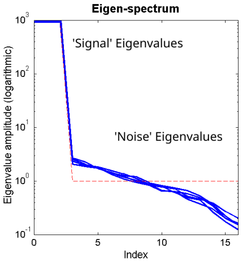
- Idea 2: Average noise eigenvalues
- Project data onto noise subspace to estimate noise power σ^2

$$\mathbf{N} = \mathbf{U_2}\mathbf{U_2}^H\mathbf{X}$$

- Calculate a σ^2 over several snapshots

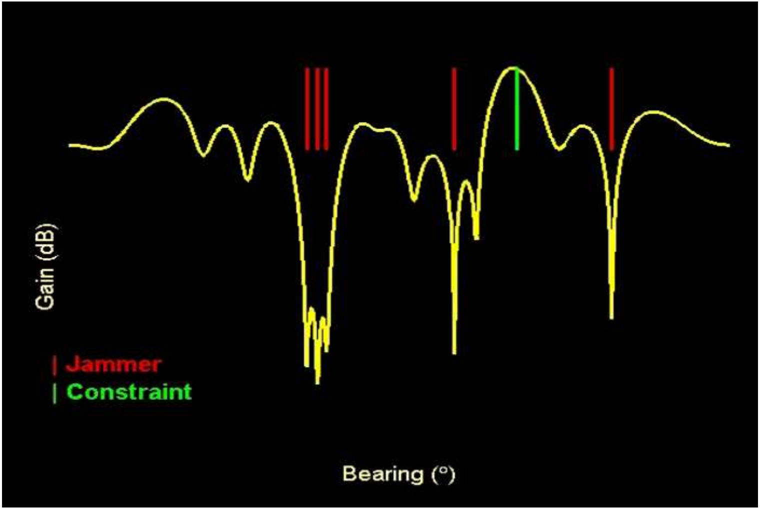
$$\hat{\mathbf{R}} = \mathbf{U}\begin{bmatrix} \mathbf{\Lambda_A} + \Lambda_{N1} & 0 \\ 0 & \sigma^2 I \end{bmatrix}\mathbf{U}^H$$

- Need to decide how to partition U into U_1 and U_2.
- Look at eigenvalues
- Can use simple thresholding or more complicated information theory.



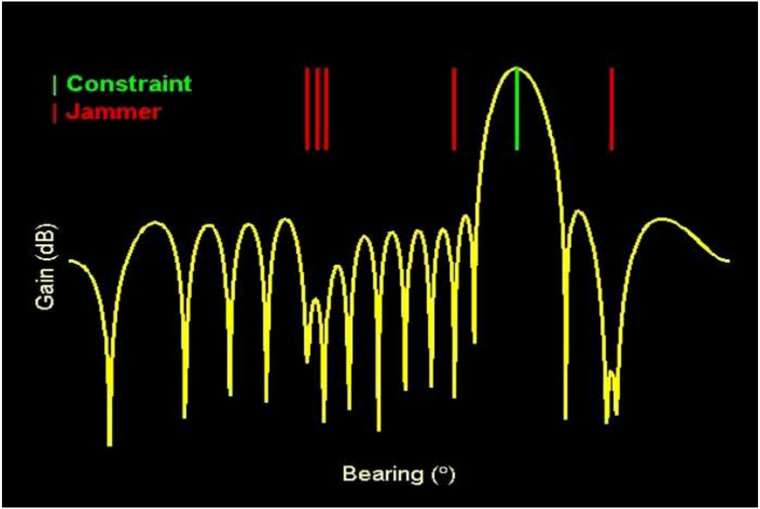
Minimum Variance Distortionless Response (MVDR)

- Recall multiple noise realizations (blocks of data) caused jitter



Stabilized MVDR Beamformer

- Average noise eigenvalues



Array Calibration Errors

- MVDR minimises power in output signal.
- $\mathbf{w} = 0$ would do this but also removes wanted signal
- 'Look direction' constraint protects the wanted signal

$$\mathbf{w}^H \mathbf{a}(\theta) = 1$$

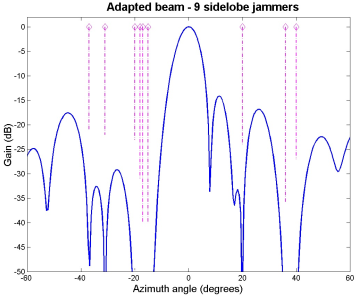
- What if $\mathbf{a}(\theta)$ is incorrect?
- Wanted signal looks like an unwanted one!
- Add extra constraints
  - More that one 'Look direction' constraint
  - Flatten main lobe – gradient constraint
- Incorporate calibration into problem and solve ...

Linearly Constrained Minimum Variance (LCMV)

- MVDR has only one constraint; can we do better?
- LCMV algorithm
  - Minimum Variance = Minimise energy of output
  - Linearly Constrained = More than one constraint (e.g. could have fixed null)

$$\mathbf{w}^H \mathbf{C} = \mathbf{g}^T$$

- Solution: $$\mathbf{w} = \mathbf{R}^{-1}\mathbf{C}\left(\mathbf{C}^H\mathbf{R}^{-1}\mathbf{C}\right)^{-1}\mathbf{g}$$



- Gain in wanted direction = 1
- Gain towards other directions = 0

Linearly Constrained Minimum Variance (LCMV)

- LCMV is a constrained minimisation problem

$$\mathbf{w} = \text{Arg Min}\left(||\mathbf{w}^H \mathbf{R} \mathbf{w}||_2^2\right) st.\mathbf{w}^H \mathbf{C} = \mathbf{g}^T$$

- If there are M constraints, M components of $\mathbf{w}$ are effectively fixed
- Thus only $N - M$ 'degrees of freedom' in the choice of $\mathbf{w}$ i.e. can only null out $N - M$ signals
- Thus have to have $N - M > 0$
- Sometimes the constraints can be linearly dependent or nearly so .....

Linearly Constrained Minimum Variance (LCMV)

- Consider

$$\mathbf{w}^H\mathbf{C} = \mathbf{g}^T$$

- or

$$\left[\mathbf{w}^H\mathbf{C} - \mathbf{g}^T\right] = \left[\mathbf{w}^H, -1\right]\left[\begin{array}{c}\mathbf{C}\\ \mathbf{g}^T\end{array}\right] = 0$$

- Take SVD

$$\left[\begin{array}{c}\mathbf{C}\\ \mathbf{g}^T\end{array}\right] = \mathbf{U\Sigma V}^H$$

- Note V is full rank so we have

$$\left[\mathbf{w}^H, -1\right]\mathbf{U\Sigma} = 0$$

University of Strathclyde Engineering

Linearly Constrained Minimum Variance (LCMV)

- Say $(M-R)$ singular values $(\mathbf{\Sigma_2})$ are small

$$\mathbf{U\Sigma} = \begin{bmatrix} \mathbf{U_1} & \mathbf{U_2} \end{bmatrix} \begin{bmatrix} \mathbf{\Sigma_1} & 0 \\ 0 & \mathbf{\Sigma_2} \end{bmatrix}$$
$$\approx \begin{bmatrix} \mathbf{U_1} & \mathbf{U_2} \end{bmatrix} \begin{bmatrix} \mathbf{\Sigma_1} & 0 \\ 0 & 0 \end{bmatrix} = \begin{bmatrix} \mathbf{U_1\Sigma_1} & 0 \end{bmatrix}$$

- Then an approximation to our problem is

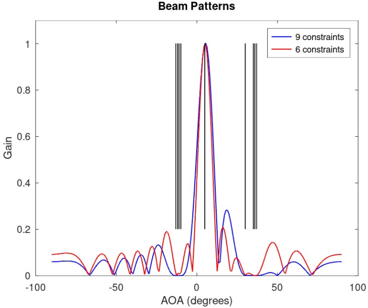
$$\left[\mathbf{w}^H, -1\right] \mathbf{U_1\Sigma_1} = 0$$

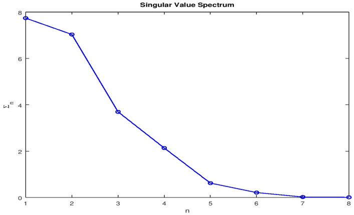
- Alternatively, writing $\mathbf{U_1\Sigma_1} = \begin{bmatrix} \tilde{\mathbf{C}} \\ \tilde{\mathbf{g}}^T \end{bmatrix}$ the approximate problems is

$$\mathbf{w}^H\tilde{\mathbf{C}} = \tilde{\mathbf{g}}^T$$

- Note $\tilde{\mathbf{C}}$ only has $R < M$ columns

Linearly Constrained Minimum Variance (LCMV)

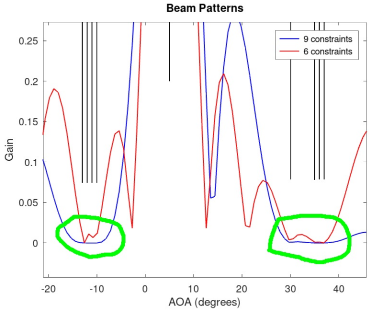




- Beam patterns
- 16 sensors
- blue - 9 Constraints
- red - 6 Constraints
- Beam patterns similar at constraint points

- Constraint matrix singular value spectrum
- 3 small singular values
- 6 constraints nearly as good as 9 constraints

Linearly Constrained Minimum Variance (LCMV)



- Beam patterns
- blue - 9 Constraints
- red - 6 Constraints

- Constraints not strictly achieved due to non-zero singular values
- Threshold on singular values should be set by acceptable ‘null’ gain

Blind Source Separation

- For adaptive beamforming, we assumed we know:
  - AOA of the wanted signal(s)
  - Array calibration
- What if we don't know this information?
  $\Rightarrow$ Blind Source Separation or Independent Component Analysis
- Assume
  - The source signals are statistically independent
  - No more than one Gaussian signal
    (higher order moments of Gaussian signal are zero)
  - Not interested in absolute amplitude of the signals
- Recall that

$$\mathbf{X} = \mathbf{A}\mathbf{S} + \mathcal{N}$$

- Define SVD of A

$$\mathbf{A} = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^H$$

- If we can calculate $\mathbf{U}$, $\mathbf{\Sigma}$ and $\mathbf{V}$ we can calculate $\mathbf{A}$

Blind Source Separation

- The covariance matrix is

$$\mathbf{R} = \mathbf{X}\mathbf{X}^H = \mathbf{A}\mathbf{D}\mathbf{A}^H + \sigma^2 I$$

- Assume that the source signals are statistically independent and have unit power i.e. $\mathbf{D} = I$.
- (If not, redefine array manifold $\mathbf{A}$ so that $\mathbf{A} \leftarrow \mathbf{A}\mathbf{D}^{\frac{1}{2}}$)
- So the covariance matrix is

$$\mathbf{R} = \mathbf{X}\mathbf{X}^H = \mathbf{A}\mathbf{A}^H + \sigma^2 I$$

- Using the SVD of $\mathbf{A}$ we find

$$\mathbf{X}\mathbf{X}^H = \mathbf{U}\left[\boldsymbol{\Sigma}\cancel{\mathbf{V}^H\mathbf{V}}\boldsymbol{\Sigma}\right]\mathbf{U}^H + \sigma^2 I \;\; = \mathbf{U}\left[\boldsymbol{\Sigma}^2 + \sigma^2 I\right]\mathbf{U}^H$$

which is the covariance matrix EVD

Blind Source Separation

- For simplicity assume $\mathbf{\Sigma}^2 + \sigma^2 I \approx \mathbf{\Sigma}^2$ i.e. high SNR then

$$\mathbf{X}\mathbf{X}^H = \mathbf{U}\mathbf{\Sigma}^2\mathbf{U}^H$$

- So the covariance matrix gives us $\mathbf{U}$ and $\mathbf{\Sigma}$, and all we have to do is to find a way to calculate $\mathbf{V}$.
- Now note that

$$\mathbf{X} = \mathbf{A}\mathbf{S} + \mathcal{N} = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^H\mathbf{S} + \mathcal{N}$$

- Thus, assuming $\mathbf{\Sigma}^{-1}$ exists,

$$\mathbf{Y} \equiv \mathbf{\Sigma}^{-1}\mathbf{U}^H\mathbf{X} = \mathbf{V}^H\mathbf{S} + \tilde{\mathcal{N}}$$

where $\tilde{\mathcal{N}} = \mathbf{\Sigma}^{-1}\mathbf{U}^H\mathcal{N}$ is a noise term

Blind Source Separation

- We have
$$\mathbf{Y} = \mathbf{V}^H \mathbf{S} + \tilde{\mathcal{N}}$$
- so $\mathbf{S}$ could be extracted from $\mathbf{Y}$ if we knew $\mathbf{V}^H$
- Then
$$\hat{\mathbf{S}} = \left(\mathbf{V}\boldsymbol{\Sigma}^{-1}\mathbf{U}^H\right)\mathbf{X}$$
- cf. bank of beamformers
$$\hat{\mathbf{S}} = \begin{bmatrix} \mathbf{w_1}^H \\ : \\ \mathbf{w_N}^H \end{bmatrix} \mathbf{X}$$
- Blind signal separation is limited by what a bank of beamformers can do e.g. N sensors $\rightarrow N-1$ nulls

Blind Source Separation

- How to estimate $\mathbf{V}^H$?
  NB
  $$\mathbf{Y}\mathbf{Y}^H = \mathbf{V}^H\mathbf{S}\mathbf{S}^H\mathbf{V} + \sigma^2\mathbf{\Sigma}^{-2}$$
  but $\mathbf{S}\mathbf{S}^H = I$ so
  $$\mathbf{Y}\mathbf{Y}^H = I + \sigma^2\mathbf{\Sigma}^{-2}$$
  i.e. not dependent on $\mathbf{V}^H$ so the second order statistics of $\mathbf{Y}$ will not help us estimate $\mathbf{V}^H$
- Can however use higher order statistics
- Can also use nonlinear cost function
- At this point we have stop manipulating matrices and deal with actual time series data

Blind Source Separation

- E.g. 'FastICA' - iteration to minimise 'negentropy'

$$J\left(Y\right) = H\left(\mathbf{Y}_{\mathbf{Gauss}}\right) - H\left(\mathbf{Y}\right)$$

- $\mathbf{Y}_{\mathbf{Gauss}}$ is Gaussian data with same covariance matrix as $\mathbf{Y}$, $H\left(\mathbf{Y}\right)$ is the entropy of $\mathbf{Y}$

$$H\left(Y\right) = -\int p_Y(y)\log(p_Y(y))dy$$

- Iteration

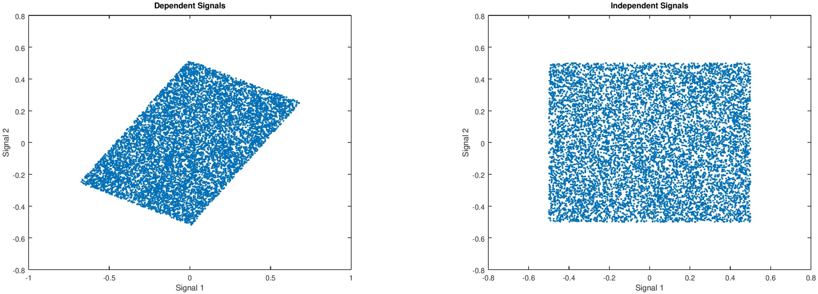
$$\mathbf{V}_{k+1} = G\left(\mathbf{V}_k^H\mathbf{Y}\right)^H\mathbf{Y} - G'\left(\mathbf{V}_k^H\mathbf{Y}\right)^H\mathbf{V}_k$$

$$G\left(v\right) = \tanh(\alpha v), v\exp(-v^2/2), \text{or } v^3$$

where $1 \leq \alpha \leq 2$

Blind Source Separation

- Can exploit higher order statistics
- Statistical independence $P(x, y) = P(x)P(y)$
- Scatter diagram



- Calculate rotation (i.e. unitary matrix $\mathbf{V}$) to align scatter plot with axes

Blind Source Separation

- Estimating the ‘hidden’ rotation matrix

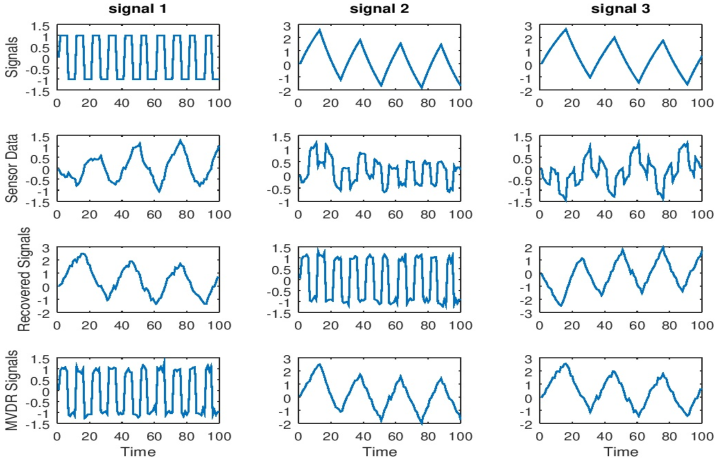
$$\mathbf{Y} = \mathbf{V}^H\mathbf{S} + \tilde{\mathcal{N}}$$

- Loop through all pairs of signals
- Rotate (with $\mathbf{Q_i}$ say) to align with axes
- Repeat until rotation angle is below a threshold

$$\mathbf{Q_n}\mathbf{Q_{n-1}}...\mathbf{Q_1}\mathbf{Y} = \hat{\mathbf{S}}$$

- Can show that $\hat{\mathbf{S}}$ is $\mathbf{S}$ up to scaling and permutation of the signal order

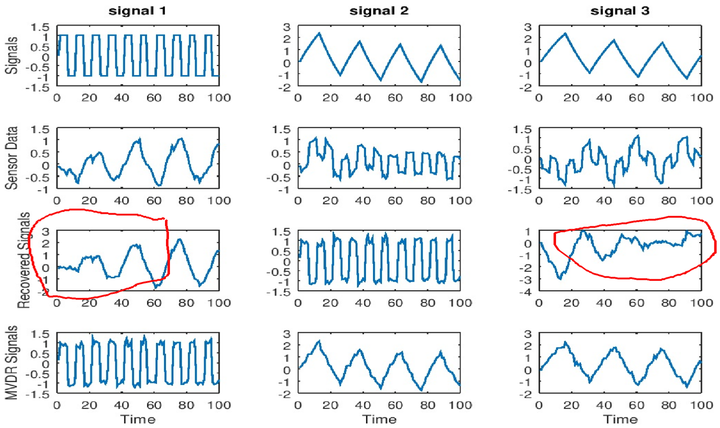
Blind Source Separation



- 3 signals,3 sensors, SNR = 20dB
- MVDR as benchmark (recall: has access to more information)

Blind Source Separation

- Need more data to calculate higher-order statistics



- Previous plot: 1000 data samples, This plot: 100 data samples
- NB. MVDR largely unaffected

Summary

- Signal Separation: need a ‘filter’ and to estimate parameters
  Performance limited by ‘optimum’ filter
- Non-adaptive beamforming
  Good optimisation algorithms
- Adaptive signal processing for beamforming
  Constrain direction of main beam, reduce everything else
  Weight jitter, calibration errors
  Lots of linear algebra
- Blind source separation Higher-order statistics or nonlinear optimisation
  Needed more data to get good result

- Acknowledgment: John Mather, QinetiQ.