Source Separation and Beamforming Background

Ian K. Proudler

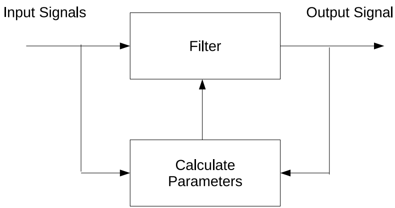
Dept. of Electronic & Electrical Engineering, Univerity of Strathclyde

UDRC Summer School, 1 July, 2021.

Source Separation and Beamforming Background: Overview

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
- We will focus on narrowband beamforming in this talk
- Broadband beamforming requires a space-time filter

Signal Separation

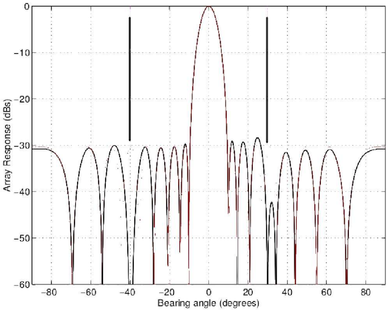
- Parameter selection – the interesting part

- 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

- Aka 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

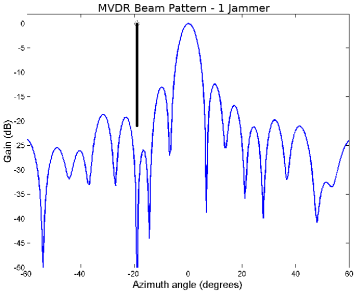
Adaptive Source Separation

- 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

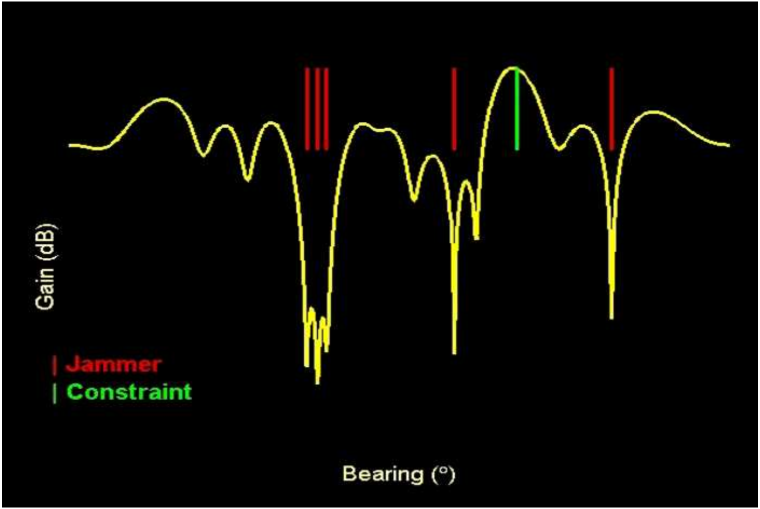
$$\mathbf{w} = \frac{\mathbf{R}^{-1}\mathbf{a}(\theta)}{\mathbf{a}^H(\theta)\mathbf{R}^{-1}\mathbf{a}(\theta)}$$



- 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

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

- "Noise" subspace manipulation
  Average noise subspace eigenvalues
- 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.

Linear Algebra

- MVDR weight vector depends on covariance matrix $\mathbf{R}$
- This matrix has structure that can be exploited
- Hermitian (symmetric)

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

- We can use linear algebra to study / manipulate the covariance matrix
- Eigenvalue decomposition of Hermitian matrix

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

- Eigenvectors: $\mathbf{U}$ is a unitary matrix

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

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

Eigenvalue Decomposition

- Eigenvectors are not steering vectors

$$\mathbf{X} = \mathbf{A}\mathbf{S}$$

- Eigenvalue Decomposition

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

- Decomposition of $\mathbf{X}$?

$$\mathbf{X} \stackrel{?}{=} \mathbf{U}\mathbf{\Lambda}^{1/2}$$

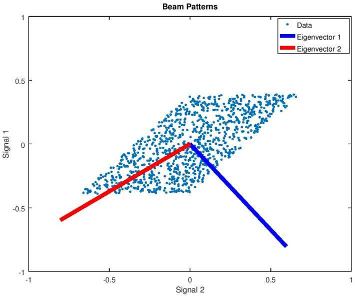
- 'Hidden' Unitary Matrix (SVD)

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

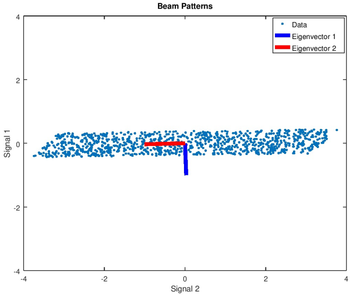
$$\mathbf{R} = \mathbf{U}\mathbf{\Lambda}^{1/2}\cancel{\mathbf{V}^H\mathbf{V}}\mathbf{\Lambda}^{1/2}\mathbf{U}^H$$

Eigenvalue Decomposition

- Eigenvectors are not steering vectors



2 equal power signals



2 signals with power ratio 10:1

- Scatter plot
- Covariance matrix EVD
- Eigenvectors approximately steering vectors when powers are dissimilar

Eigenvalue Decomposition

University of Strathclyde Engineering

- 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

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

$$\mathbf{A} = \begin{bmatrix} \mathbf{a}(\theta_1) & \mathbf{a}(\theta_2) \end{bmatrix} \qquad \mathbf{D} = \begin{bmatrix} P_1 & 0 \\ 0 & P_2 \end{bmatrix}$$

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

$$\mathbf{A}\mathbf{D}\mathbf{A}^H = \mathbf{U} \begin{bmatrix} \mathbf{\Lambda}_\mathbf{A} & 0 \\ 0 & 0 \end{bmatrix} \mathbf{U}^H$$

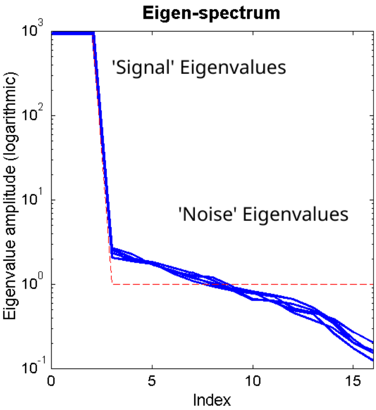
- Covariance matrix EVD

$$\mathbf{R} = \mathbf{U} \begin{bmatrix} \mathbf{\Lambda}_\mathbf{A} & 0 \\ 0 & 0 \end{bmatrix} \mathbf{U}^H + \sigma^2 I = \mathbf{U} \begin{bmatrix} \mathbf{\Lambda}_\mathbf{A} + \sigma^2 I & 0 \\ 0 & \sigma^2 I \end{bmatrix} \mathbf{U}^H$$

Eigenvalue Spectrum

- Eigenvalue spectrum

$$\begin{bmatrix} \mathbf{\Lambda}_{\mathbf{A}} + \sigma^2 \\ \sigma^2 I \end{bmatrix}$$



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

Signal and Noise Subspaces

- 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

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

- Covariance matrix EVD

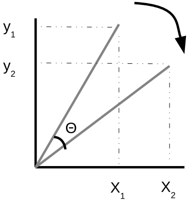
$$\mathbf{R} = \mathbf{U_1} \left(\mathbf{\Lambda_A} + \mathbf{\Lambda_{N1}} \right) \mathbf{U_1}^H + \mathbf{U_2} \left(\mathbf{\Lambda_{N2}} \right) \mathbf{U_2}^H$$

Rotation Matrices

- Eigenvectors: $\mathbf{U}$ is a unitary matrix

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

- Can be considered as a rotation in N-dimensional space
- 2-D case (Givens rotations)



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

- Can build N-D rotation from 2-D ones

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

Singular Value Decomposition

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

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

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

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

  Eigenvalues are the square of the singular values

Stabilized MVDR Beamformer

- Recall basic MVDR beamformer suffers from weight jitter
- Covariance matrix EVD

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

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

$$\hat{\mathbf{X}} = \mathbf{U_1}\mathbf{U_1}^H\mathbf{X}$$

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

- Issues with rank deficient $\hat{\mathbf{R}}$

Stabilized MVDR Beamformer

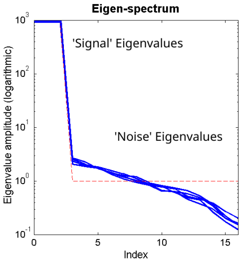
- Average noise eigenvalues
- Project data onto noise subspace

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

- Calculate a σ over several snapshots

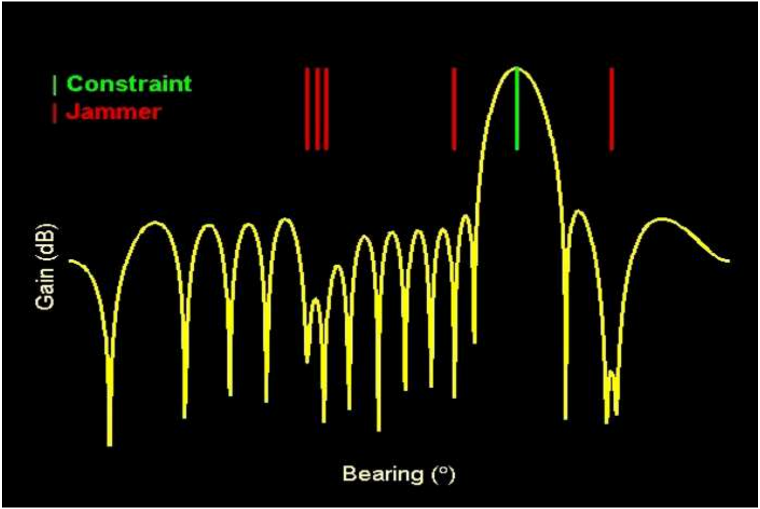
$$\hat{\mathbf{R}} = \mathbf{U}\begin{bmatrix} \mathbf{\Lambda_A} + \Lambda_{N1} & 0 \\ 0 & \sigma^2 I \end{bmatrix}\mathbf{U}^H$$

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



Stabilized MVDR Beamformer

- Average noise eigenvalues



Array Calibration Errors

- MVDR minimises power in output signal.
- $\mathbf{w} = 0$ would do this
- '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 ...

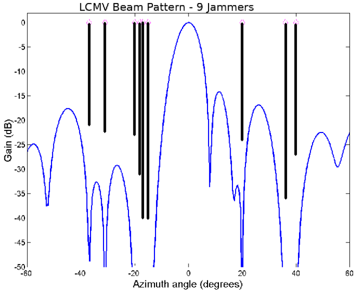
Linearly Constrained Minimum Variance (LCMV)

- Minimum Variance = Minimise energy of output
- Linearly Constrained = More than one constraint

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

- 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

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

V is full rank so

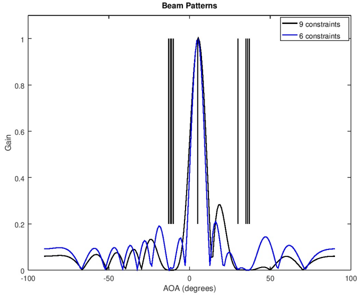
$$\left[\mathbf{w}^H, -1\right] U\Sigma = 0$$

If $N - R$ singular values are small

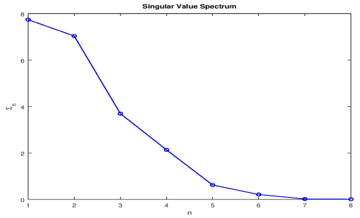
$$\left[\mathbf{w}^H, -1\right] U_1\Sigma_1 = 0$$

Let $U_1\Sigma_1 = \left[\begin{array}{c}\tilde{\mathbf{C}}\\ \tilde{\mathbf{g}}^T\end{array}\right]$ then $\mathbf{w}^H\tilde{\mathbf{C}} = \tilde{\mathbf{g}}^T$ and $\tilde{\mathbf{C}}$ only has R columns

Linearly Constrained Minimum Variance (LCMV)

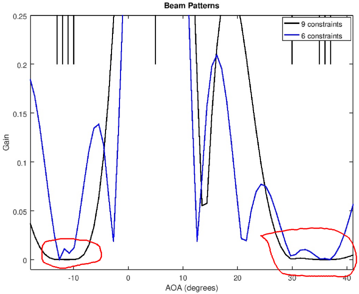


- Beam patterns
- black - 9 Constraints
- blue - 6 Constraints
- Beam patterns similar at constraint points



- Constraint matrix singular value spectrum
- 3 small singular values
- 6 constraints $\approx$ 9 constraints

Linearly Constrained Minimum Variance (LCMV)



- Beam patterns
- black - 9 Constraints
- blue - 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

- What if we don't know AOA of wanted signal and array calibration?
- Recall that

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

- Covariance matrix

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

  Assume that the source signals are statistically independent an unit power i.e. $\mathbf{D} = I$. If not redefine array manifold $\mathbf{A}$ so that $\mathbf{A} \leftarrow \mathbf{A}\mathbf{D}^{\frac{1}{2}}$

- Define SVD of A

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

Blind Source Separation

- Covariance matrix EVD

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

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

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

- So the covariance matrix gives us $\mathbf{U}$ and $\mathbf{\Sigma}$. Now note that

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

- Thus

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

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

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

$$\hat{\mathbf{S}} = \left(\mathbf{V}\mathbf{\Sigma}^{-1}\mathbf{U}^H\right)\mathbf{X}$$

- cf. bank of beamformers

$$\hat{\mathbf{S}} = \begin{bmatrix} \mathbf{w_1}^H \\ \vdots \\ \mathbf{w_N}^H \end{bmatrix} \mathbf{X}$$

- 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. a diagonal matrix – 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

Blind Source Separation

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

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

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

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

- Iteration

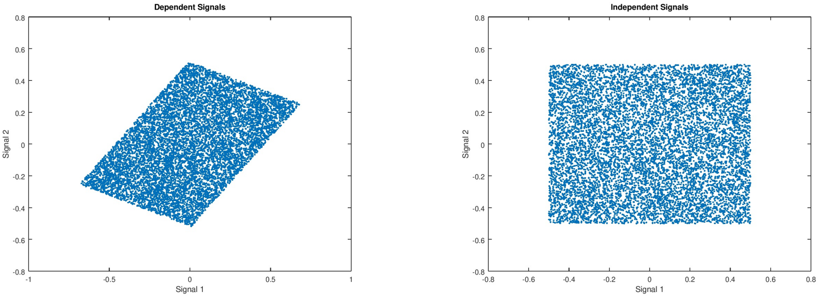
$$\mathbf{V}_{k+1} = G\left(\mathbf{V}_k^H\mathbf{Y}\right)^H\mathbf{Y} - G'\left(\mathbf{V}_k^H\mathbf{Y}\right)^H\mathbf{V}_k$$

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

where $1 \leq \alpha \leq 2$

Blind Source Separation

- 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

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

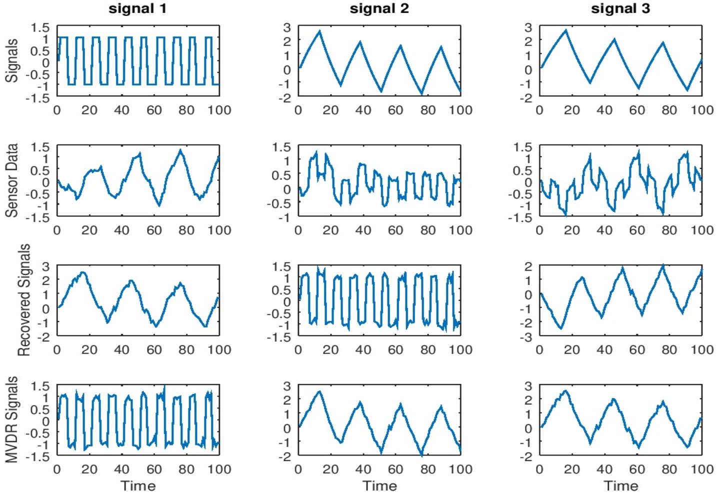
- Loop through all pairs of signals
- Rotate to align with axes
- Repeat until rotation angle is below a threshold

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

i.e. $J\left(\hat{\mathbf{S}}\right) < \epsilon$ but $J\left(\mathbf{S}\right) = 0$ so $\hat{\mathbf{S}} \approx \mathbf{S}$

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

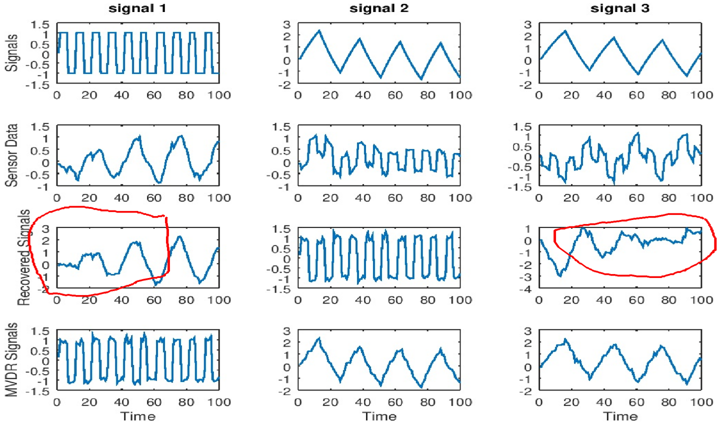
Blind Source Separation



- 3 signals,3 sensors, SNR = 20dB, MVDR as benchmark

Blind Source Separation

- Need more data to calculate higher-order statistics



- Previous plot: 1000 data samples, This plot: 100 data samples

Summary

- Signal Separation: filter and 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
  Lots of data needed

- Acknowledgment: John Mather (QinetiQ).