Spatial Smoothing and Broadband Beamforming

Bhaskar D Rao
University of California, San Diego
Email: brao@ucsd.edu

Reference Books and Papers

1. Optimum Array Processing, H. L. Van Trees
2. Stoica, P., & Moses, R. L. (2005). Spectral analysis of signals (Vol. 1). Upper Saddle River, NJ: Pearson Prentice Hall.
3. Schmidt, R. (1986). Multiple emitter location and signal parameter estimation. IEEE transactions on antennas and propagation, 34(3), 276-280.
4. Roy, Richard, and Thomas Kailath. "ESPRIT-estimation of signal parameters via rotational invariance techniques." IEEE Transactions on acoustics, speech, and signal processing 37.7 (1989): 984-995.
5. Rao, Bhaskar D., and K. S. Arun. "Model based processing of signals: A state space approach." Proceedings of the IEEE 80.2 (1992): 283-309.

Narrow-band Signals

$$\mathbf{x}(\omega_c, n) = \mathbf{V}(\omega_c, \mathbf{k}_s)F_s[n] + \sum_{l=1}^{D-1} \mathbf{V}(\omega_c, \mathbf{k}_l)F_l[n] + \mathbf{Z}[n]$$

Assumptions

- $F_s[n]$, $F_l[n], l = 1, .., D-1$, and $\mathbf{Z}[n]$ are zero mean
- $E(|F_s[n]|^2) = p_s$, and $E(|F_l[n]|^2 = p_l, l = 1, \ldots, D-1$, and $E(\mathbf{Z}[n]\mathbf{Z}^H[n]) = \sigma_z^2\mathbf{I}$
- All the signals/sources are uncorrelated with each other and over time: $E(F_l[n]F_m^*[p]) = p_l\delta[l-m]\delta[n-p]$ and $E(F_l[n]F_s^*[p]) = 0$
- The sources are uncorrelated with the noise: $E(\mathbf{Z}[n]F_l^*[m]) = \mathbf{0}$

Direction of Arrival (DOA) Estimation

$$\mathbf{x}[n] = \sum_{l=0}^{D-1} \mathbf{V}_l F_l[n] + \mathbf{Z}[n], n = 0, 1, \ldots, L-1$$

Here we have chosen to denote $\mathbf{V}_s$ by $\mathbf{V}_0$ for simplicity of notation. Note $\mathbf{V}_l = \mathbf{V}(\mathbf{k}_l)$. We will refer to the L vector measurements as L snapshots.

Goal: Estimate the direction of arrival $\mathbf{k}_l$ or equivalently $(\theta_l, \phi_l), l = 0, ..., D-1$.
Can also consider estimating the source powers $p_l, l = 0, 1, ..., D-1$, and noise variance σ_z^2.

ULA :
$\mathbf{V}(\psi) = [e^{-j\frac{N-1}{2}\psi}, \ldots, e^{j\frac{N-1}{2}\psi}]^T = e^{-j\frac{N-1}{2}\psi}[1, e^{j\psi}, e^{j2\psi}, ..., e^{j(N-1)\psi}]^T$
Note for a ULA, $\mathbf{V}(\psi) = \mathbf{J}\mathbf{V}^*(\psi)$, where $\mathbf{J}$ is all zeros except for ones along the anti-diagonal. Also note $\mathbf{J}^2 = \mathbf{I}$.

ULA and Time Series

ULA

$$\begin{bmatrix} x_0[n] \\ x_1[n] \\ \vdots \\ x_{N-1}[n] \end{bmatrix} = \sum_{l=0}^{D-1} \begin{bmatrix} 1 \\ e^{j\psi_l} \\ \vdots \\ e^{j(N-1)\psi_l} \end{bmatrix} e^{-j\frac{N-1}{2}\psi_l} F_l[n] + \mathbf{Z}[n], n = 0, 1, \ldots, L-1$$

Similar to a time series problems composed of complex exponentials $x[n] = \sum_{l=0}^{D-1} c_l e^{j\omega_l n} + z[n], n = 0, 1, \ldots, (M-1)$.
In vector form, the time series measurement is

$$\begin{bmatrix} x[0] \\ x[1] \\ \vdots \\ x[M-1] \end{bmatrix} = \sum_{l=0}^{D-1} \begin{bmatrix} 1 \\ e^{j\omega_l} \\ \vdots \\ e^{j(M-1)\omega_l} \end{bmatrix} c_l + \begin{bmatrix} z[0] \\ z[1] \\ \vdots \\ z[M-1] \end{bmatrix}$$

The sum of exponentials problem with M samples is equivalent to a ULA with M sensors and one (vector) measurement or one snapshot
In array processing we have a limited number of sensors N but can compensate potentially with L snapshots.
The time series problem is equivalent to one snapshot but a potentially large array if the number of samples M is large

Observations

$$
\begin{aligned}
\mathbf{S}_x &= \mathbf{S}_s + \sigma_z^2 \mathbf{I} \\
&= \mathbf{E}_s \left(\Lambda_s + \sigma_z^2 \mathbf{I}_{D \times D} \right) \mathbf{E}_s^H + \sigma_z^2 \mathbf{E}_n \mathbf{E}_n^H \\
&= \sum_{l=1}^{D} (\lambda_l^s + \sigma_z^2) \mathbf{q}_l \mathbf{q}_l^H + \sigma_z^2 \sum_{l=(D+1)}^{N} \mathbf{q}_l \mathbf{q}_l^H
\end{aligned}
$$

- $\mathbf{S}_x$ has eigenvalues $\lambda_l = \lambda_l^s + \sigma_z^2, l = 1, \ldots, D$, and $\lambda_l = \sigma_z^2, l = (D+1), \ldots, N$
- The number of sources D can be identified by examining the multiplicity of the smallest eigenvalue which is expected to be σ_z^2 with multiplicity $(N - D)$
- The eigenvectors corresponding to the signal subspaces and noise subspaces are readily available from the eigen decomposition of $\mathbf{S}_x$, i.e. $\mathbf{E}_s$ and $\mathbf{E}_n$ are easily extracted.

MUltiple SIgnal Classification (MUSIC)

MUSIC uses the noise subspace spanned by the "noise" eigenvectors $\mathbf{E}_n$.

Since $\mathcal{R}(\mathbf{V}) = \mathcal{R}(\mathbf{E}_s) \perp \mathcal{R}(\mathbf{E}_n)$, we have
$\mathbf{V}_l \perp \mathcal{R}(\mathbf{E}_n), l = 0, 1, \ldots, (D-1)$ or $\mathbf{V}_l^H \mathbf{E}_n = 0, l = 0, 1, \ldots, (D-1)$
Many options to use this orthogonality property. All methods start with $\mathbf{S}_x$, usually estimated from the data.

MUSIC Algorithm (also known as Spectral-MUSIC)

- Perform an eigen-decomposition of $\mathbf{S}_x$
- Estimate D and the signal subspace $\mathbf{E}_s$ and the noise subspace $\mathbf{E}_n$
- Compute the spatial MUSIC spectrum as follows:

$$P_{MUSIC}(\mathbf{k}) = \frac{1}{|\mathbf{V}^H(\mathbf{k})\mathbf{E}_n|^2}$$

  Note that $|\mathbf{V}^H(\mathbf{k})\mathbf{E}_n|^2 = \mathbf{V}^H(\mathbf{k})\mathbf{E}_n\mathbf{E}_n^H\mathbf{V}(\mathbf{k}) = \mathbf{V}^H(\mathbf{k})\mathbf{P}_n\mathbf{V}(\mathbf{k})$ where $\mathbf{P}_n = \mathbf{E}_n\mathbf{E}_n^H$

- The peaks in $P_{MUSIC}(\mathbf{k})$ gives us the DOA.

Root-MUSIC Algorithm for ULAs

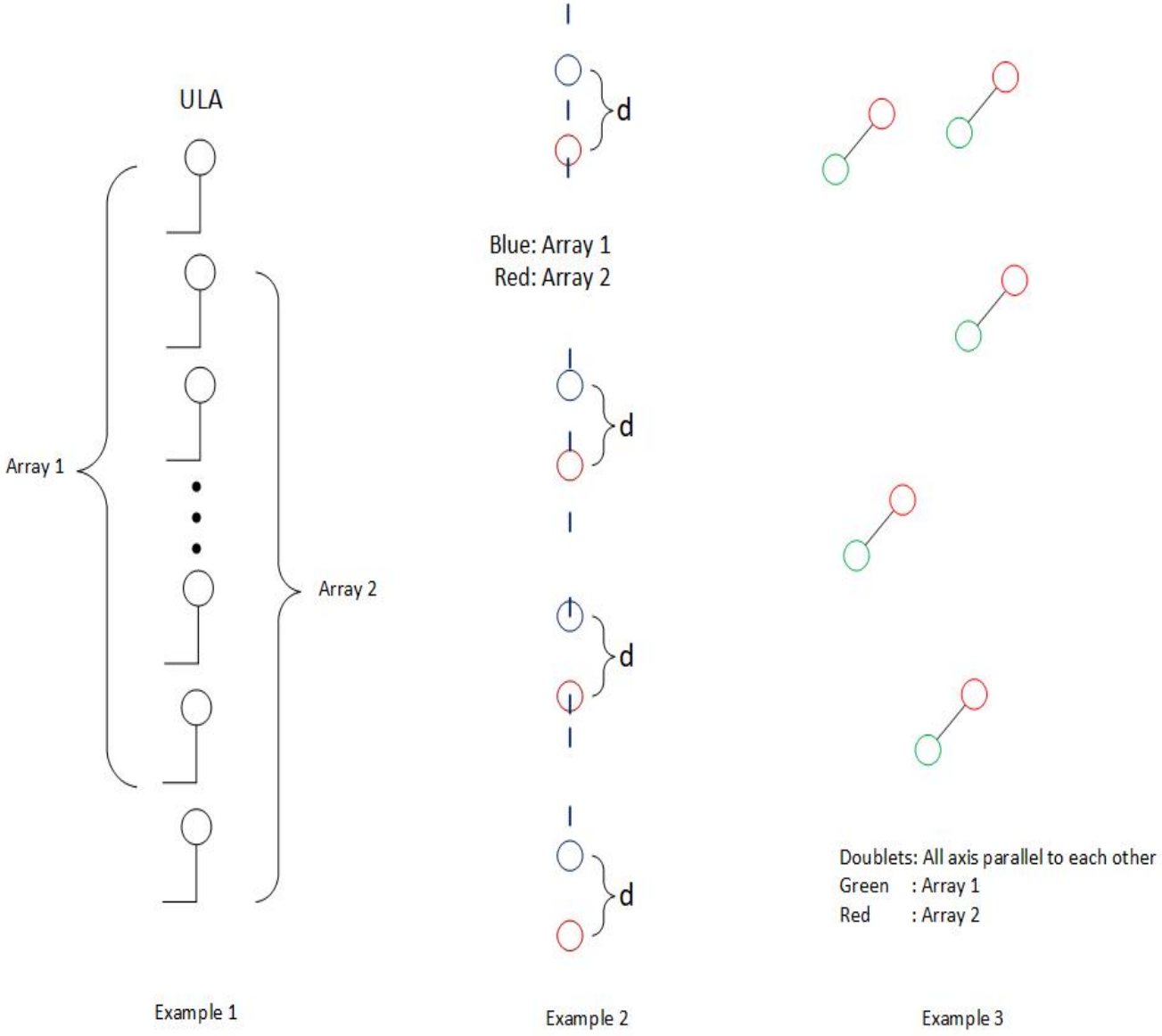
- Perform an eigen-decomposition of $\mathbf{S}_x$
- Estimate D and the signal subspace $\mathbf{E}_s$ and the noise subspace $\mathbf{E}_n$
- Compute $q_l(z) = \sum_{m=0}^{N-1} q_l[m]z^{-m}, l = D+1, \ldots, N-1$.
- Compute $B(z) = \sum_{l=D+1}^{N} B_l(z)$ where $B_l(z) = \mathbf{q}_l(z)\mathbf{q}_l^*(\frac{1}{z^*}) = \sum_{l=-(N-1)}^{N-1} r_{q_l}[m]z^{-m}$
- Perform a spectral factorization to find a minimum phase factor $H_{rm}(z)$, i.e. $B(z) = H_{rm}(z)H_{rm}^*(\frac{1}{z^*})$.
- The D roots on/close to the unit circle are identified and their argument used to estimate the DOA

ESPRIT algorithm

- Perform an eigen-decomposition of $\mathbf{S}_x$
- Estimate D and the signal subspace $\mathbf{E}_s$ and the noise subspace $\mathbf{E}_n$
- From $\mathbf{E}_s$ determine $\mathbf{E}_1$ as the first $(N-1)$ rows and $\mathbf{E}_2$ as the last $(N-1)$ rows of $\mathbf{E}_s$
- Compute Φ by solving $\mathbf{E}_2 = \mathbf{E}_1\Phi$ using a Least Squares Approach, i.e. $\Phi = \mathbf{E}_1^+\mathbf{E}_2$.
- Perform an eigen-decomposition of Φ
- The argument of the D eigenvalues are used to estimate the DOA

As in the minimum norm method, $\mathbf{E}_1^+$ can be obtained efficiently using the matrix inversion lemma and the computation of Φ can be simplified. Another option is to use a Total Least Squares approach, as opposed to a Least Squares approach, to estimate Φ

ESPRIT: ULA and Doublets



ESPRIT with Doublets: Theory

If the array manifold for the first array is $\mathbf{v}_1(\mathbf{k})$, and the array manifold for the second array is $\mathbf{v}_2(\mathbf{k})$,then for a plane with wavenumber $\mathbf{k}_l$, we have $\mathbf{v}_2(\mathbf{k}_l) = \mathbf{v}_1(\mathbf{k}_l)e^{-j\omega_c\tau_l}$. The overall array manifold has the form

$$\mathbf{V}(\mathbf{k}_l) = \begin{bmatrix} \mathbf{v}_1(\mathbf{k}_l) \\ \mathbf{v}_2(\mathbf{k}_l) \end{bmatrix} = \begin{bmatrix} \mathbf{v}_1(\mathbf{k}_l) \\ \mathbf{v}_1(\mathbf{k}_l)e^{-j\omega_c\tau_l} \end{bmatrix}$$

Now consider D plane waves. The signal subspace

$$\mathbf{V} = [\mathbf{V}_0, \mathbf{V}_1, \ldots, \mathbf{V}_{D-1}] = \begin{bmatrix} \mathbf{v}_1(\mathbf{k}_0) & \mathbf{v}_1(\mathbf{k}_1) & \ldots & \mathbf{v}_1(\mathbf{k}_{D-1}) \\ \mathbf{v}_2(\mathbf{k}_0) & \mathbf{v}_2(\mathbf{k}_1) & \ldots & \mathbf{v}_2(\mathbf{k}_{D-1}) \end{bmatrix} = \begin{bmatrix} V_1 \\ V_2 \end{bmatrix}$$

$$= \begin{bmatrix} \mathbf{v}_1(\mathbf{k}_0) & \mathbf{v}_1(\mathbf{k}_1) & \ldots & \mathbf{v}_1(\mathbf{k}_{D-1}) \\ \mathbf{v}_1(\mathbf{k}_l)e^{-j\omega_c\tau_0} & \mathbf{v}_1(\mathbf{k}_l)e^{-j\omega_c\tau_1} & \ldots & \mathbf{v}_1(\mathbf{k}_l)e^{-j\omega_c\tau_{D-1}} \end{bmatrix} = \begin{bmatrix} V_1 \\ V_1\Psi \end{bmatrix}$$

An ESPRIT algorithm can be developed now with

$$\mathbf{E}_s = \begin{bmatrix} \mathbf{E}_1 \\ \mathbf{E}_2 \end{bmatrix} \text{ with } \mathbf{E}_2 = \mathbf{E}_1\Phi$$

Coherent Sources

$$\mathbf{S}_x = \mathbf{S}_s + \sigma_z^2 \mathbf{I} = \mathbf{V}\mathbf{S}_F\mathbf{V}^H + \sigma_z^2 \mathbf{I}$$

The dimension D of the signal subspace depends on the rank of the source covariance matrix $\mathbf{S}_F$.

If the sources are highly correlated, $\mathbf{S}_F$ can be ill-conditioned or singular. This can make the dimension of the signal subspace less than D and in particular when an estimate of $\mathbf{S}_x$ is used.

Spatial Smoothing for ULAs: A technique to overcome this rank deficiency problem.

Spatial Smoothing

We will consider the ULA as two sub-arrays as in ESPRIT: subarray 1 is the first $(N-1)$ sensors and subarray 2 is the last $(N-1)$ sensors. Since the data is given by $\mathbf{x}[n] = \mathbf{V}\mathbf{F}[n] + \mathbf{Z}[n]$, we have the data from the two sub-arrays given by

$$\mathbf{x}_1[n] = \mathbf{V}_1\mathbf{F}[n] + \mathbf{Z}_1[n] \text{ and } \mathbf{x}_2[n] = \mathbf{V}_2\mathbf{F}[n] + \mathbf{Z}_2[n]$$

Covariance of the two subarrays are given by

$$\mathbf{S}_x^{(1)} = \mathbf{V}_1\mathbf{S}_F\mathbf{V}_1^H + \sigma_z^2\mathbf{I} \text{ and } \mathbf{S}_x^{(2)} = \mathbf{V}_2\mathbf{S}_F\mathbf{V}_2^H + \sigma_z^2\mathbf{I} = \mathbf{V}_1\Psi\mathbf{S}_F\Psi^H\mathbf{V}_1^H + \sigma_z^2\mathbf{I}$$

Spatially Smoothed Covariance

$$\begin{aligned}
\mathbf{S}_x^{(s)} &= \frac{1}{2}\left(\mathbf{S}_x^{(1)} + \mathbf{S}_x^{(2)}\right) = \frac{1}{2}\left(\mathbf{V}_1\mathbf{S}_F\mathbf{V}_1^H + \sigma_z^2\mathbf{I} + \mathbf{V}_2\mathbf{S}_F\mathbf{V}_2^H + \sigma_z^2\mathbf{I}\right) \\
&= \frac{1}{2}\left(\mathbf{V}_1\mathbf{S}_F\mathbf{V}_1^H + \mathbf{V}_1\Psi\mathbf{S}_F\Psi^H\mathbf{V}_1^H\right) + \sigma_z^2\mathbf{I} \\
&= \mathbf{V}_1\left(\frac{\mathbf{S}_F + \Psi\mathbf{S}_F\Psi^H}{2}\right)\mathbf{V}_1^H = \mathbf{V}_1\tilde{\mathbf{S}}_F\mathbf{V}_1^H
\end{aligned}$$

$\tilde{\mathbf{S}}_F = \frac{\mathbf{S}_F + \Psi\mathbf{S}_F\Psi^H}{2}$ is more likely to have rank D restoring the signal subspace dimension in the spatially smoothed covariance matrix

Spatial Smoothing Cont'd

We can generalize this idea by viewing the ULA as P sub-arrays: subarray 1 is the first $(N-P+1)$ sensors, and subarray 2 is the next $(N-P+1)$ sensors and so on.
Since the data is given by $\mathbf{x}[n] = \mathbf{V}\mathbf{F}[n] + \mathbf{Z}[n]$, we have the data from the l sub-array given by

$$\mathbf{x}_l[n] = \mathbf{V}_l\mathbf{F}[n] + \mathbf{Z}_l[n], l = 1, 2, .., P \text{ with } \mathbf{V}_l = \mathbf{V}_1\Psi^{l-1}$$

Covariance of the lth sub-array is given by

$$\mathbf{S}_x^{(l)} = \mathbf{V}_l\mathbf{S}_F\mathbf{V}_l^H + \sigma_z^2\mathbf{I} = \mathbf{V}_1\Psi^{l-1}\mathbf{S}(\Psi^H)^{l-1}\mathbf{V}_1 + \sigma_z^2\mathbf{I}$$

Spatially Smoothed Covariance

$$\mathbf{S}_x^{(s)} = \frac{1}{P}\sum_{l=1}^{P}\mathbf{S}_x^{(l)} = \mathbf{V}_1\tilde{\mathbf{S}}_F\mathbf{V}_1^H$$

$\tilde{\mathbf{S}}_F = \frac{1}{P}\sum_{l=1}^{P}\Psi^{l-1}\mathbf{S}_F(\Psi^H)^{l-1}$ is more likely to have rank D restoring the signal subspace dimension in the spatially smoothed covariance matrix

Covariance Estimation

Given L snapshots

$$\hat{\mathbf{S}}_x = \frac{1}{L}\sum_{n=0}^{L-1}\mathbf{x}[n]\mathbf{x}^H[n] = \frac{1}{L}DD^H, \quad \text{where} \quad D = [\mathbf{x}[0], \mathbf{x}[1], \ldots, \mathbf{x}[L-1]]$$

Smoothed Covariance Estimate

$$\hat{\mathbf{S}}_x^{(s)} = \frac{1}{P}\sum_{l=1}^{P}\hat{\mathbf{S}}_x^{(l)} = \frac{1}{PL}\sum_{l=1}^{P}D_l D_l^H \quad \text{where} \quad D_l = [\mathbf{x}_l[0], \mathbf{x}_l[1], \ldots, \mathbf{x}_l[L-1]]$$

and $\mathbf{x}_l[n] = [x_{l-1}[n], x_l[n], \ldots, x_{N-P+l-1}[n]]^T$.
For $P = 2$, we have $\mathbf{x}_1[n] = [x_0[n], x_1[n], \ldots, x_{N-2}[n]]^T$ and $\mathbf{x}_2[n] = [x_1[n], x_2[n], \ldots, x_{N-l}[n]]^T$.
Alternatively, we can express the smoothed covariance estimate as

$$\hat{\mathbf{S}}_x^{(s)} = \frac{1}{L}\sum_{n=0}^{L-1}D_s[n]D_s^H[n], \quad \text{where} \quad D_s[n] = [\mathbf{x}_1[n], \mathbf{x}_2[n], \ldots, \mathbf{x}_P[n]]$$

Time Series: Sum of Exponentials

$x[n], n = 0, 1, \ldots, M-1$: $L = 1$, one snapshot, array size $N = M$, where M is the number of samples containing D exponentials.
To compute a covariance matrix, we have to do smoothing.

$$D_s = \begin{bmatrix} x[0] & x[1] & \ldots & x[P-1] \\ x[1] & x[2] & \ldots & x[P] \\ \vdots & \vdots & \vdots & \vdots \\ x[M-P] & x[M-P+1] & \ldots & x[M-1] \end{bmatrix}$$

The smoothed covariance can be computed a $\hat{\mathbf{S}}_x^{(s)} = \frac{1}{P} D_s D_s^H$.
Since we need rank of the matrix to be greater than D, we need $P > D$ and $M - P + 1 > D$. This requires $M > 2D$.
Instead of an eigen-decomposition of $\hat{\mathbf{S}}_x^{(s)}$, one can do a SVD of D_s.
Numerically superior.
Can use MUSIC, ESPRIT or any subspace method to estimate the frequencies ω_l.

Forward-Backward (FB) Smoothing for ULAs

For a ULA, we have $\mathbf{J}\mathbf{V}^*(\psi) = \mathbf{V}(\psi)$. Hence

$$\mathbf{J}\mathbf{S}_x^*\mathbf{J} = \mathbf{J}\mathbf{V}^*\mathbf{S}_F^*\mathbf{V}^T\mathbf{J} + \sigma_z^2\mathbf{J}\mathbf{I}\mathbf{J} = \mathbf{V}\mathbf{S}_F^*\mathbf{V}^H + \sigma_z^2\mathbf{I}$$

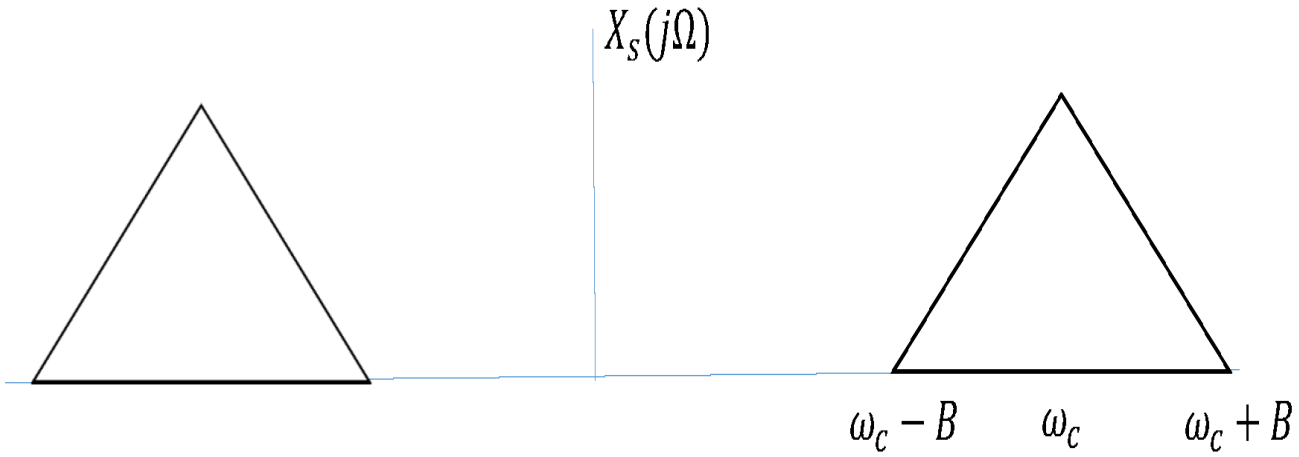
The signal subspace of $\mathbf{J}\mathbf{S}_x^*\mathbf{J}$ is also spanned by $\mathbf{V}$ and so subspace methods MUSIC etc can be applied to $\mathbf{J}\mathbf{S}_x^*\mathbf{J}$.
The FB covariance estimate

$$\mathbf{S}_x^{FB} = \frac{1}{2}\left(\mathbf{S}_x + \mathbf{J}\mathbf{S}_x^*\mathbf{J}\right) = \frac{1}{2}\mathbf{V}(\mathbf{S}_F + \mathbf{S}_F^*)\mathbf{V}^H + \sigma_z^2\mathbf{I}$$

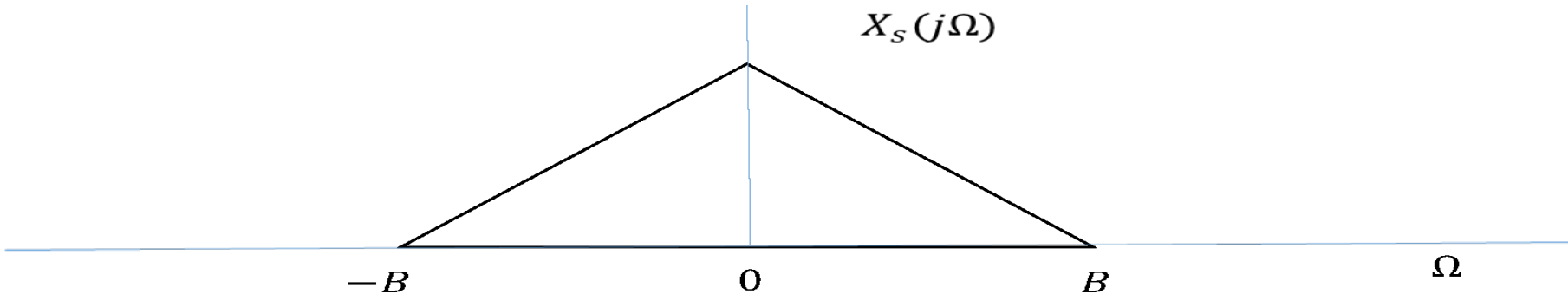
Subspace methods can be readily applied to estimated $\hat{\mathbf{S}}_x^{FB}$. The FB covariance matrix has a better condition number and so improves performance.
The FB approach can be applied to spatial smoothing, i.e. $\frac{1}{2}\left(\hat{\mathbf{S}}_x^{(s)} + \mathbf{J}\hat{\mathbf{S}}_x^{*(s)}\mathbf{J}\right)$ and also to the time series $[D_s, \mathbf{J}D_s^*]$.

Broadband Signals

Narrow band signals: $\tilde{f}(t-\tau)e^{j\omega_c(t-\tau)} \approx \tilde{f}(t)e^{j\omega_c(t-\tau)}$, for $\tau << T = \frac{1}{2B}$

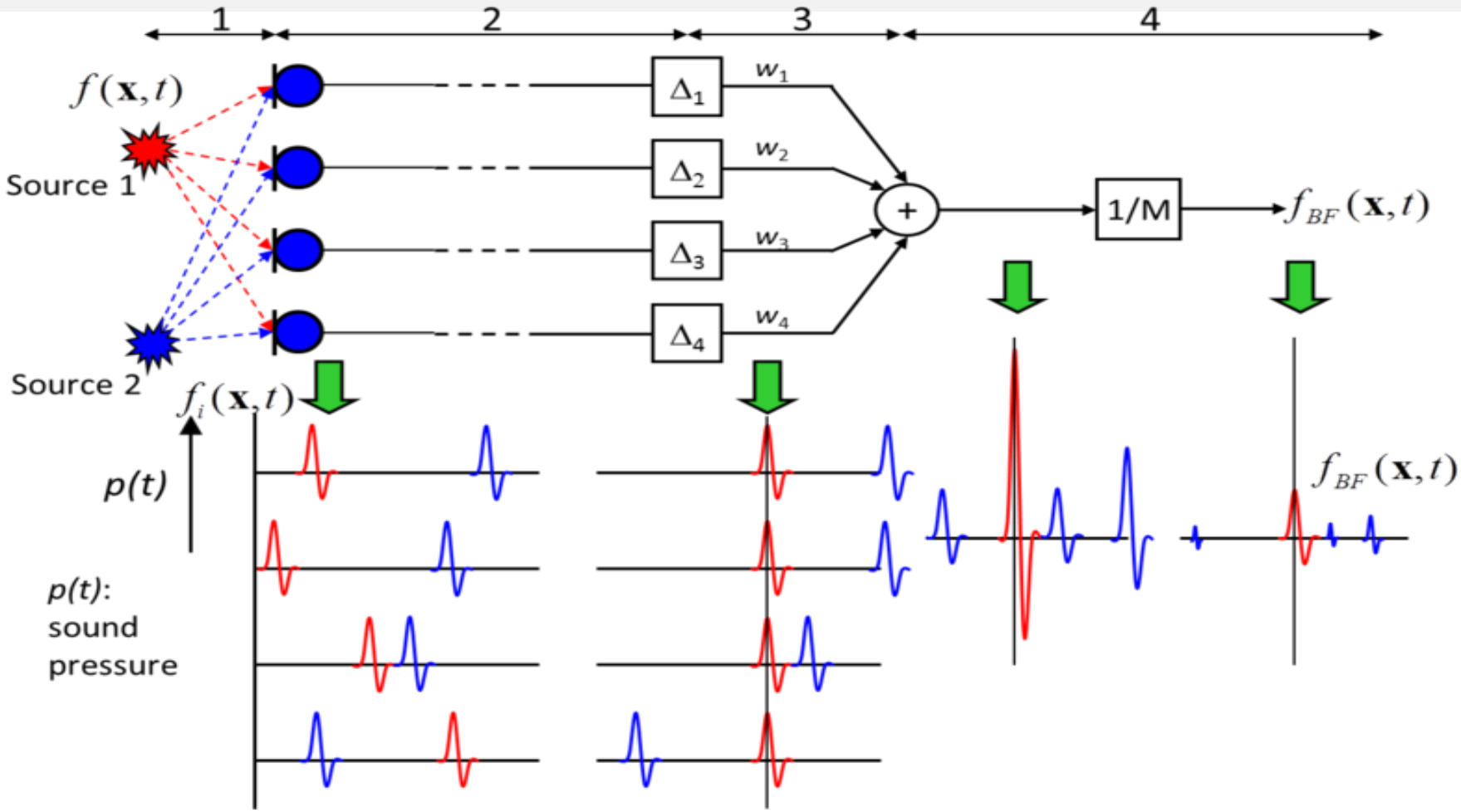


The signal becomes broadband if the bandwidth becomes large and the delay across the array can no longer be considered small compared to the sampling interval



Signals such as speech signals are naturally baseband signals and are broadband in nature

Delay and Sum beamformer



Delay the output of the sensors appropriately and enable constructive addition of the copies to get SNR improvement over white noise
In the presence of an interferer will use FIR filters at the output to cancel or suppress interference

Time-Delay Estimation: Preliminaries[1]

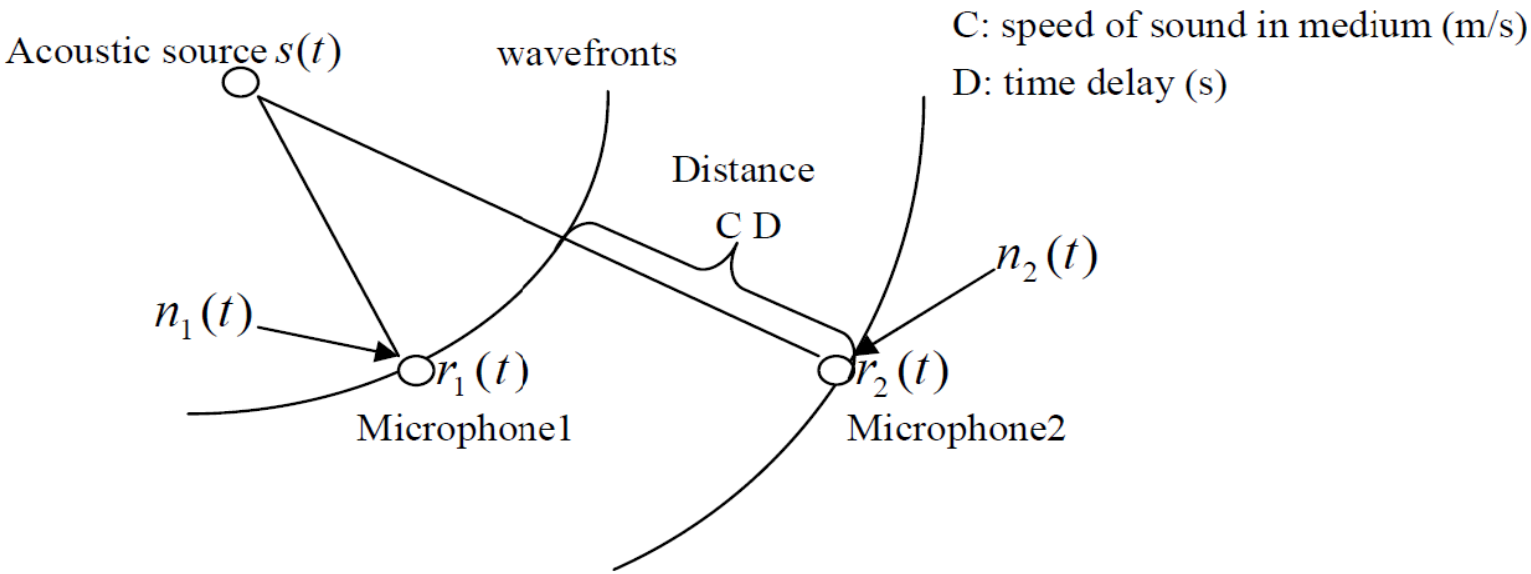
$r_{x_1x_2}[m] = E(x_1[n+m]x_2^*[n])$ and from LTI systems theory we know the cross-power spectrum is given by

$$S_{y_1y_2}(e^{j\omega}) = \sum_{m=-\infty}^{\infty} r_{y_1y_2}[m]e^{-j\omega m} = H_1(e^{j\omega})H_2^*(e^{j\omega})S_{x_1x_2}(e^{j\omega})$$

$S_{y_1y_1}(e^{j\omega})$ is the power spectrum which is real and positive.
If we are given data, we can divide the data into L blocks of length N and estimate the cross-power spectrum using the FFT

[1] Carter, G. Clifford. Coherence and time delay estimation: an applied tutorial for research, development, test, and evaluation engineers. IEEE, 1993.

Time Delay Estimation



$$
\begin{aligned}
y_1[n] &= r_1(nT) = s[n] + z_1[n] \\
y_2[n] &= r_2(nT) = s[n-D] + z_2[n]
\end{aligned}
$$

Assuming the noises are uncorrelated, $r_{y_2 y_1}[m] = r_{ss}[m-D]$.

$$
\hat{D} = \arg\max_m r_{y_2 y_1}[m] = \arg\max_m \frac{1}{2\pi} \int_{-\pi}^{\pi} S_{y_2 y_1}(e^{j\omega}) e^{j\omega m} d\omega
$$

The peak is sharp if $s[n]$ is a white noise sequence and in the absence of reverberation/multipath.

Generalized Cross-Correlation (GCC) Methods

$$\hat{D} = \arg\max_m \frac{1}{2\pi} \int_{-\pi}^{\pi} \phi(e^{j\omega}) S_{y_2 y_1}(e^{j\omega}) e^{j\omega m} d\omega$$

Cross-Correlation Method: $\phi(e^{j\omega}) = 1$
The purpose of $\phi(e^{j\omega})$ is to whiten the signal and/or weight the spectrum based on the SNR in the frequency domain. Note that $S_{y_2 y_1}(e^{j\omega}) = S_{ss}(e^{j\omega}) e^{-j\omega D}$.
PHAse Transform (PHAT): $\phi(e^{j\omega}) = \frac{1}{|S_{y_2 y_1}(e^{j\omega})|}$
Smoothed COherence Transform (SCOT): $\phi(e^{j\omega}) = \frac{1}{\sqrt{S_{y_1 y_1}(e^{j\omega}) S_{y_2 y_2}(e^{j\omega})}}$
Maximum-Likelihood (ML): $\phi(e^{j\omega}) = \frac{1}{|S_{y_2 y_1}(e^{j\omega})|} \frac{|\gamma_{y_2 y_1}(e^{j\omega})|^2}{1 - |\gamma_{y_2 y_1}(e^{j\omega})|^2}$ where

$$\gamma_{y_2 y_1}(e^{j\omega}) = \frac{S_{y_2 y_1}(e^{j\omega})}{\sqrt{S_{y_1 y_1}(e^{j\omega}) S_{y_2 y_2}(e^{j\omega})}}, 0 \leq |\gamma_{y_2 y_1}(e^{j\omega})| \leq 1, \forall \omega$$

is the coherence function.

Adaptive Broadband Beamformer[2]

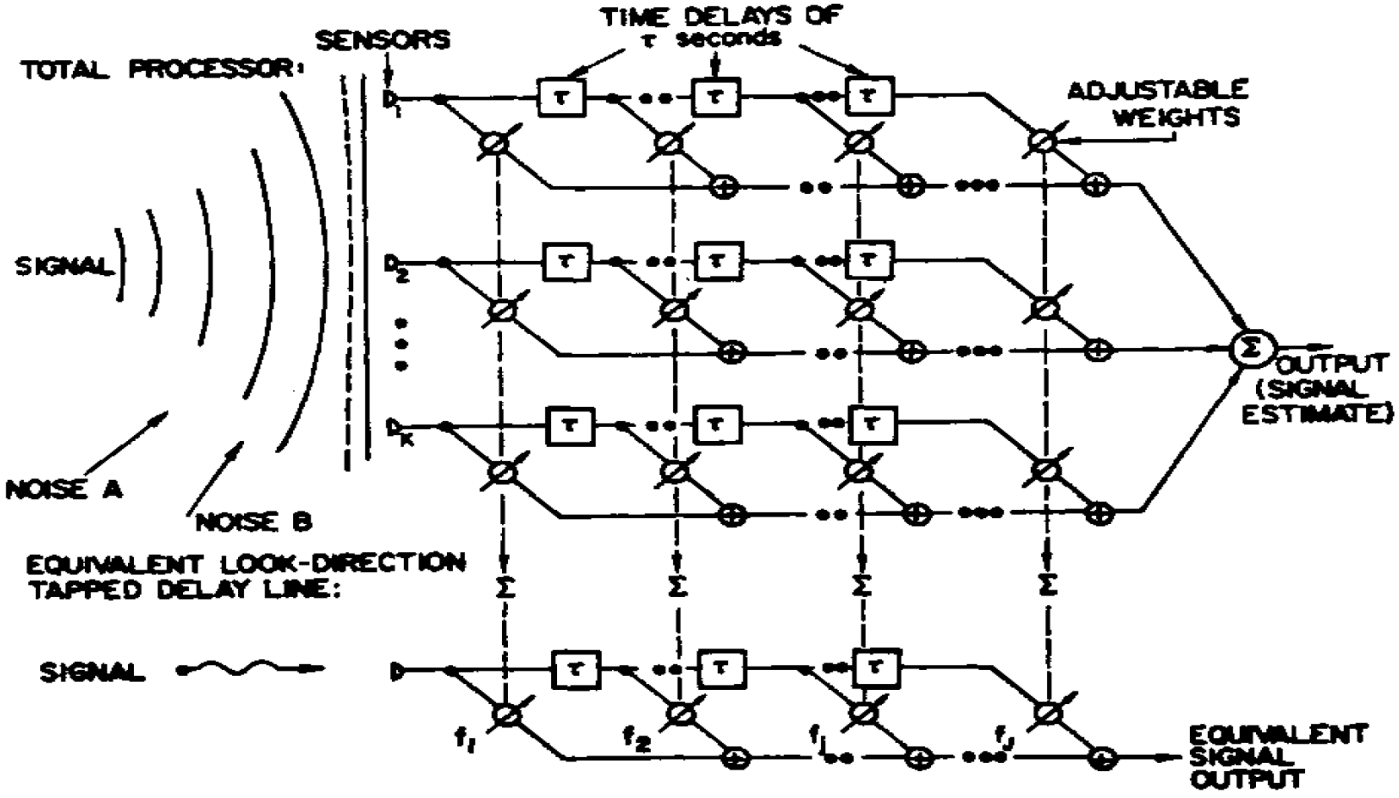


Fig. 1. Broad-band antenna array and equivalent processor for signals coming from the look direction.

[2]Frost, Otis Lamont. "An algorithm for linearly constrained adaptive array processing." Proceedings of the IEEE 60.8 (1972): 926-935.

Wideband BF Formulation

Number of sensors: $N = K$, number of taps, i.e. order of FIR filters is J.

If we denote the input measurement vector as $\mathbf{x}[n]$, then the data in the tap delay line is $[\mathbf{x}[n], \mathbf{x}[n-1], \ldots, \mathbf{x}[n-J+1]]$

Weights: Using vector notation, the weights in the tap delay line are $[W_0, W_1, \ldots, W_{J-1}]$, where $W_k = [w_{k,0}, w_{k,1}, \ldots, w_{k,N-1}]$.

Key Question: How to characterize and design such an array?

Characterization: Frequency-Wavenumber Response

Extension of the MPDR formulation: Look direction constraint plus minimize output power

Frequency Wavenumber Response

If the source signal is $s(t) = e^{j\Omega t}$ from direction $\mathbf{k}$, then the signal received at sensor l is delayed relative to the origin by τ_l.

$$x_l(t) = s(t) = e^{j\Omega(t-\tau_l)} \text{ or } x_l[n] = x_l(nT) = e^{j\Omega(nT-\tau_l)} = e^{j\omega(n-\tilde{\tau}_l)}$$

where $\omega = \Omega T$ and $\tilde{\tau}_l = \frac{1}{T}\tau_l$.

$$x_l[n] \to H_l(z) \to y_l[n] = e^{j\omega(n-\tilde{\tau}_l)} H_l(e^{j\omega})$$

The array output

$$y[n] = \sum_{l=0}^{N-1} y_l[n] = \sum_{l=0}^{N-1} e^{j\omega(n-\tilde{\tau}_l)} H_l(e^{j\omega}) = e^{j\omega n} \sum_{l=0}^{N-1} e^{j\omega\tilde{\tau}_l} H_l(e^{j\omega}) = e^{j\omega n} B(\omega, \mathbf{k})$$

$B(\omega, \mathbf{k}) = \sum_{l=0}^{N-1} e^{j\omega\tilde{\tau}_l} H_l(e^{j\omega})$ is the frequency-wavenumber response of array

Time-Domain Formulation

Beamformer output

$$y[n] = W_0^T \mathbf{x}[n] + W_1^T \mathbf{x}[n-1] + \ldots + W_{J-1}^T \mathbf{x}[n-J+1] = \sum_{k=0}^{J-1} W_k^T \mathbf{x}[n-k]$$

Constraints: Define $\mathbf{1} = [1, 1, \ldots, 1]^T$, a vector in R^N.

$$\sum_{m=0}^{N-1} w_{k,m} = f_k \text{ or } \mathbf{1}^T W_k = f_k, k = 0, 1, \ldots, J-1$$

This is equivalent to look direction transfer function constraint of

$$H_d(z) = F(z) = f_0 + f_1 z^{-1} + \ldots + f_{J-1} z^{-(J-1)} = \sum_{m=0}^{J-1} f_m z^{-m}$$

Frost Beamformer

Stacking all the BF weights and the constraints into large vectors of dimension NJ, we have

$$y[n] = \mathbf{W}^T\mathbf{X}[n], \text{ where } \mathbf{W} = [W_0^T, W_1^T, \ldots, W_{J-1}^T]^T$$

and $\mathbf{X}[n] = [\mathbf{x}^T[n], \mathbf{x}^T[n-1], \ldots, \mathbf{x}^T[n-J+1]]^T$.
The output power is $E(y^2[n]) = \mathbf{W}^T\mathbf{R}_{xx}\mathbf{W}$ where $\mathbf{R}_{xx} = E(\mathbf{X}[n]\mathbf{X}^T[n])$.
The constraints are given by

$$\mathbf{C}^T\mathbf{W} = \mathbf{f} \text{ or } \begin{bmatrix} \mathbf{1}^T & \mathbf{0} & \ldots & \mathbf{0} \\ \mathbf{0} & \mathbf{1}^T & \ldots & \mathbf{0} \\ \vdots & \vdots & \vdots & \vdots \\ \mathbf{0} & \mathbf{0} & \ldots & \mathbf{1}^T \end{bmatrix} \mathbf{W} = \begin{bmatrix} f_1 \\ f_2 \\ \vdots \\ f_{J-1} \end{bmatrix}$$

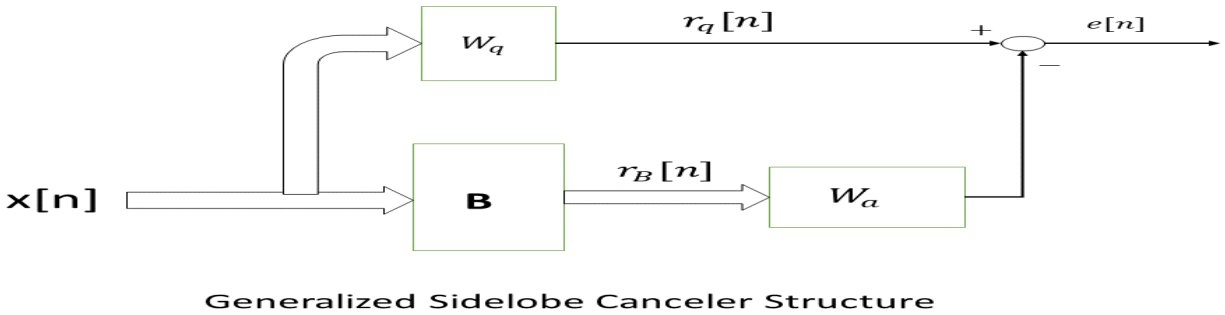
The optimization problem is

$$\min_{\mathbf{W}} \mathbf{W}^T\mathbf{R}_{xx}\mathbf{W} \text{ subject to } \mathbf{C}^T\mathbf{W} = \mathbf{f}$$

Using Lagrange multipliers, we can show

$$\mathbf{W}_o = \mathbf{R}_{xx}^{-1}\mathbf{C}(\mathbf{C}^T\mathbf{R}_{xx}^{-1}\mathbf{C})^{-1}\mathbf{f}$$

GSC structure



Because of the symmetric nature of the constraints with respect to time $\mathbf{1}^T W_k = f_k$, i.e. the constraint matrix $\mathbf{1}$ is the same, the constraint can be implemented first followed by filtering. Our $W_q = \frac{1}{N}[1, 1, \ldots, 1]^T$ and our B matrix is an orthonormal set of vectors orthogonal to W_q. If $N = 8$, a choice for B is from the Haar basis

$$B = \begin{bmatrix} 1 & 0 & 0 & 0 & 1 & 0 & 1 \\ -1 & 0 & 0 & 0 & 1 & 0 & 1 \\ 0 & 1 & 0 & 0 & -1 & 0 & 1 \\ 0 & -1 & 0 & 0 & -1 & 0 & 1 \\ 0 & 0 & 1 & 0 & 0 & 1 & -1 \\ 0 & 0 & -1 & 0 & 0 & 1 & -1 \\ 0 & 0 & 0 & 1 & 0 & -1 & -1 \\ 0 & 0 & 0 & -1 & 0 & -1 & -1 \end{bmatrix}$$

GSC Implementation of Adaptive Broadband BF[3]

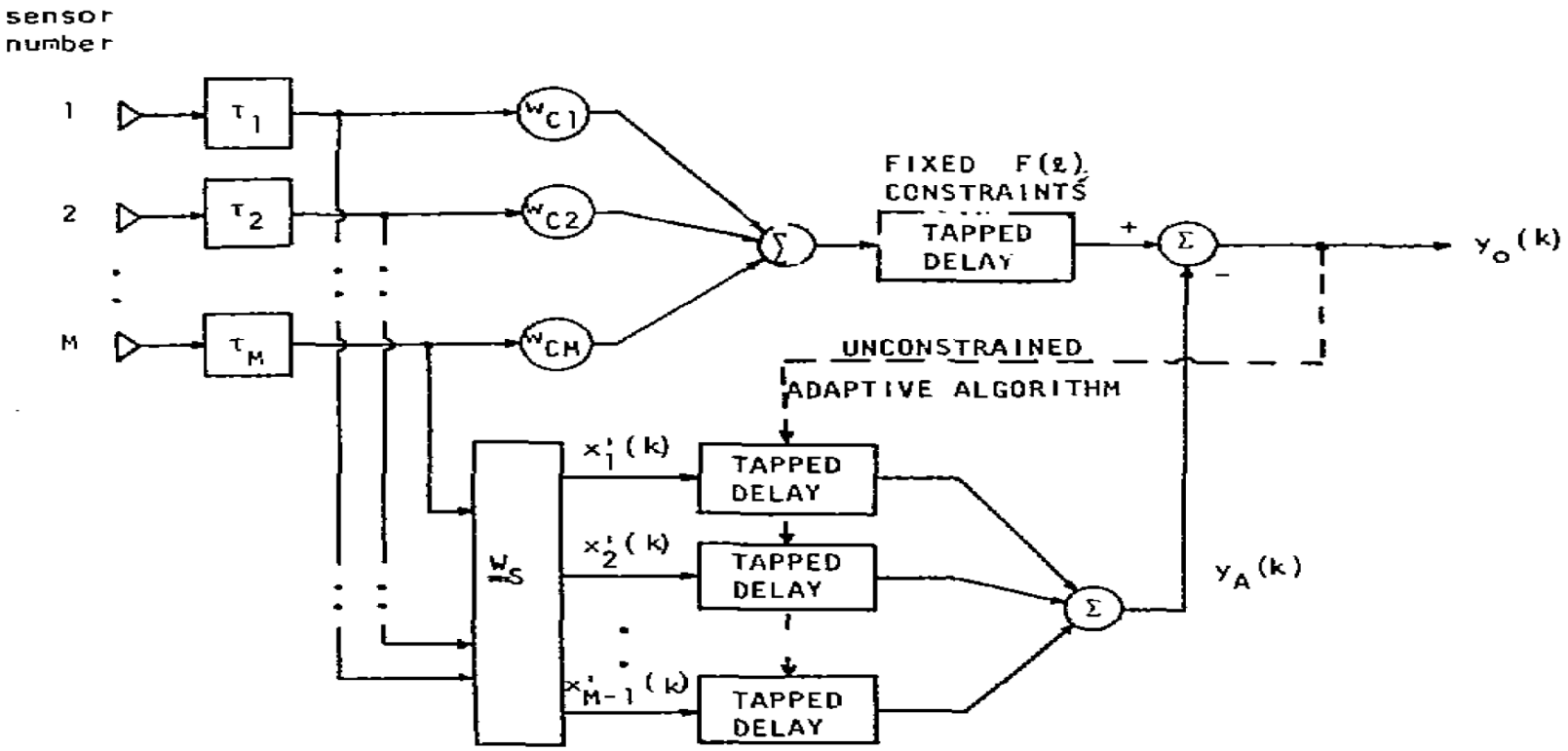


Fig. 4. Generalized sidelobe canceling form of linearly constrained adaptive array processing algorithm.

[3]Griffiths, Lloyd, and C. W. Jim. "An alternative approach to linearly constrained adaptive beamforming." IEEE Transactions on antennas and propagation 30.1 (1982): 27-34.