Adaptive and Array Signal Processing

1. Complex Analysis

1.1. Derivatives of Non-analytic functions

$$h:\mathbb{C}\ni z\mapsto h(z)\in\mathbb{C}$$
$$f:\mathbb{R}^2\ni(x,y)\mapsto f(x,y)\in\mathbb{C},\ z=x+jy$$
$$g:\mathbb{C}^2\ni(z_1,z_2)\mapsto g(z_1,z_2)\in\mathbb{C},\ x=\frac{z+z^*}{2},\ y=\frac{z-z^*}{2\mathrm{j}}$$

Derivative in direction φ:
$\frac{\mathrm{d}h}{\mathrm{d}z}=\left(\frac{\partial f}{\partial x}\cos(\varphi)+\frac{\partial f}{\partial y}\sin\varphi\right)\mathrm{e}^{-\mathrm{j}\varphi},\ \mathrm{d}z=\mathrm{e}^{\mathrm{j}\varphi}\mathrm{d}t,\quad \varphi,\mathrm{d}t\in\mathbb{R}$

$$\frac{\partial g}{\partial z}=\frac{1}{2}\left(\frac{\partial f}{\partial x}-\mathrm{j}\frac{\partial f}{\partial y}\right),\quad \frac{\partial g}{\partial z^*}=\frac{1}{2}\left(\frac{\partial f}{\partial x}+\mathrm{j}\frac{\partial f}{\partial y}\right)$$

1.2. Analytic functions

Definition:

$\frac{\mathrm{d}h}{\mathrm{d}z}$ independent of φ
$$\forall(x,y)\in\mathbb{R}^2:\quad \frac{\partial f}{\partial x}+\mathrm{j}\frac{\partial f}{\partial y}=0$$
$$\frac{\mathrm{d}h}{\mathrm{d}z}=\frac{\partial f}{\partial x}=-\mathrm{j}\frac{\partial f}{\partial y}$$
$$\frac{\partial g}{\partial z^*}\equiv 0$$

- Ansatz for obtaining the Lemma: compute derivative of $\frac{\mathrm{d}h}{\mathrm{d}z}$ wrt. to φ and set it to 0.
- If $g()$ depends on z^* it is not analytic

1.3. Minimization of $h(z)=g(z,z^\star)\in\mathbb{R}$

Necessary condition for an extremum:
$$\frac{\partial g}{\partial z^*}=0$$

Direction of steepest descent: $z\leftarrow z-\mu\frac{\partial g}{\partial z^\star},\quad \mu>0$

Useful derivatives:
$$\frac{\partial\left(\boldsymbol{z}^{\mathrm{H}}\boldsymbol{p}+\boldsymbol{p}^{\mathrm{H}}\boldsymbol{z}\right)}{\partial\boldsymbol{z}^\star}=\boldsymbol{p}$$
$$\frac{\partial\left(\boldsymbol{z}^{\mathrm{H}}\boldsymbol{R}\boldsymbol{z}\right)}{\partial\boldsymbol{z}^*}=\boldsymbol{R}\boldsymbol{z}$$
$$\frac{\partial\,\mathrm{tr}\left(\boldsymbol{S}^{\mathrm{H}}\boldsymbol{B}\right)}{\partial\boldsymbol{S}^*}=\boldsymbol{B}$$

1.4. Quadratic minimization with linear equality constraints

Problem:
$\min_{\boldsymbol{z}}\boldsymbol{z}^{\mathrm{H}}\boldsymbol{R}\boldsymbol{z},\quad$ such that $\quad\boldsymbol{A}^{\mathrm{H}}\boldsymbol{z}=\boldsymbol{b},\quad \boldsymbol{R}=\boldsymbol{R}^{\mathrm{H}}>\boldsymbol{0}$

Corresponding Lagrange-ian function:
$\mathcal{L}=\boldsymbol{z}^{\mathrm{H}}\boldsymbol{R}\boldsymbol{z}+\boldsymbol{\lambda}^{\mathrm{H}}\left(\boldsymbol{A}^{\mathrm{H}}\boldsymbol{z}-\boldsymbol{b}\right)+\left(\boldsymbol{z}^{\mathrm{H}}\boldsymbol{A}-\boldsymbol{b}^{\mathrm{H}}\right)\boldsymbol{\lambda}$

Resulting dual optimization problem: $\min_z\max_\lambda\mathcal{L}$

Solution:
$$\boldsymbol{z}_{\mathrm{opt}}=\boldsymbol{R}^{-1}\boldsymbol{A}\left(\boldsymbol{A}^{\mathrm{H}}\boldsymbol{R}^{-1}\boldsymbol{A}\right)^{-1}\boldsymbol{b}$$
$$\min\boldsymbol{z}^H\boldsymbol{R}\boldsymbol{z}=\boldsymbol{b}^H(\boldsymbol{A}^H\boldsymbol{R}^{-1}\boldsymbol{A})^{-1}\boldsymbol{b}$$

1.5. Real-valued quadratic minimization with linear inequality constraints

Problem:
$\min_{\boldsymbol{x}}\boldsymbol{x}^{\mathrm{T}}\boldsymbol{C}\boldsymbol{x},\quad$ subject to $\quad\boldsymbol{A}\boldsymbol{x}\leq\boldsymbol{b}$

Corresponding Lagrange-ian function:
$\mathcal{L}=\boldsymbol{x}^{\mathrm{T}}\boldsymbol{C}\boldsymbol{x}+\boldsymbol{\lambda}^{\mathrm{T}}(\boldsymbol{A}\boldsymbol{x}-\boldsymbol{b})$
Resulting dual optimization problem: $\min\max\mathcal{L}$ subject to $\lambda\geq 0$

Algorithm for determining the solution:

$$\min_{\boldsymbol{x}}\ \boldsymbol{x}^{\mathrm{T}}\boldsymbol{C}\boldsymbol{x},\quad\text{subject to}\quad \boldsymbol{A}\boldsymbol{x}\leq\boldsymbol{b},\quad\text{where }\boldsymbol{C}=\boldsymbol{C}^{\mathrm{T}}>\boldsymbol{0}.$$

1. $\boldsymbol{E}\leftarrow\frac{1}{4}\boldsymbol{A}\boldsymbol{C}^{-1}\boldsymbol{A}^{\mathrm{T}}\in\mathbb{R}^{M\times M}$
2. $\lambda_k\leftarrow 0,\ $ for $k\in\{1,2,\dots,M\}$.
3. **repeat**
   for $k=1:M$ **do**
   $$\lambda_k\leftarrow\max\left(0\,,\ \frac{-1}{E_{k,k}}\left(\sum_{n=1}^{k-1}E_{k,n}\lambda_n+\frac{b_k}{2}+\sum_{n=k+1}^{M}E_{k,n}\lambda_n\right)\right)$$
   end
   until negligible change in all λ_k.
4. $\boldsymbol{x}\leftarrow-\frac{1}{2}\boldsymbol{C}^{-1}\boldsymbol{A}^{\mathrm{T}}\boldsymbol{\lambda}$

2. Linear Algebra

2.1. Vector Space

A complex vector space $\mathcal{V}$ is a set with the following properties:

1. $\forall\boldsymbol{a},\boldsymbol{b}\in\mathcal{V}:\quad\boldsymbol{a}+\boldsymbol{b}\in\mathcal{V}$
2. $\forall\boldsymbol{a},\boldsymbol{b}\in\mathcal{V}:\quad\boldsymbol{a}+\boldsymbol{b}=\boldsymbol{b}+\boldsymbol{a}$
3. $\forall\boldsymbol{a},\boldsymbol{b},\boldsymbol{c}\in\mathcal{V}:\quad(\boldsymbol{a}+\boldsymbol{b})+\boldsymbol{c}=\boldsymbol{a}+(\boldsymbol{b}+\boldsymbol{c})$
4. $\forall\boldsymbol{a}\in\mathcal{V}:\exists\boldsymbol{0}\in\mathcal{V}:\quad\boldsymbol{a}+\boldsymbol{0}=\boldsymbol{a}$
5. $\forall\boldsymbol{a}\in\mathcal{V}:\exists-\boldsymbol{a}\in\mathcal{V}:\quad\boldsymbol{a}+(-\boldsymbol{a})=\boldsymbol{0}$
6. $\forall\boldsymbol{a}\in\mathcal{V}:\quad 1\boldsymbol{a}=\boldsymbol{a}$
7. $\forall\boldsymbol{a}\in\mathcal{V},\forall\lambda,\mu\in\mathbb{C}:\quad\lambda(\mu\boldsymbol{a})=(\lambda\mu)\boldsymbol{a}$
8. $\forall\boldsymbol{a},\boldsymbol{b}\in\mathcal{V},\forall\lambda\in\mathbb{C}:\quad\lambda(\boldsymbol{a}+\boldsymbol{b})=\lambda\boldsymbol{a}+\lambda\boldsymbol{b}$
9. $\forall\boldsymbol{a}\in\mathcal{V},\forall\lambda,\mu\in\mathbb{C}:\quad(\lambda+\mu)\boldsymbol{a}=\lambda\boldsymbol{a}+\mu\boldsymbol{a}$

2.2. Linear Subspace

A set $\mathcal{S}$ is called a subspace of a complex vector space $\mathcal{V}$ iff:

1. $\mathcal{S}\subseteq\mathcal{V}$
2. $\forall\boldsymbol{a},\boldsymbol{b}\in\mathcal{V}:\quad\boldsymbol{a}+\boldsymbol{b}=\boldsymbol{b}+\boldsymbol{a}$
3. $\forall\boldsymbol{a},\boldsymbol{b}\in\mathcal{S}:\quad\boldsymbol{a}+\boldsymbol{b}\in\mathcal{S}$
4. $\forall\boldsymbol{a}\in\mathcal{S},\forall\lambda\in\mathbb{C}:\quad\lambda\boldsymbol{a}\in\mathcal{S}$

2.3. Linear (In)dependence

The vectors $\boldsymbol{v}_1,\dots,\boldsymbol{v}_n\in\mathcal{V}$ are said to be linearly independent iff:
$\sum_{k=1}^n a_k\boldsymbol{v}_k=\boldsymbol{0}\implies a_1=\dots=a_n=0$

The vectors are linearly dependent iff:

$\exists i:\exists b_1,\dots,b_{i-1},b_{i+1},\dots,b_n\in\mathbb{C}:\quad\boldsymbol{v}_i=\sum_{k=1,k\neq i}^{n}b_k\boldsymbol{v}_k$

- $\boldsymbol{v}_1,\dots,\boldsymbol{v}_n\in\mathcal{V}$ are LI, and $\boldsymbol{s}\in\mathcal{V}$ cannot be expressed as a linear combination, then $\boldsymbol{v}_1,\dots,\boldsymbol{v}_n,\boldsymbol{s}$ are LI
- $\dim(\mathcal{S})$ of a subspace is the maximum number of LI vectors that fit in it
- For every subspace $\mathcal{S}$, with $\dim(\mathcal{S})=n$, and any LI vectors $\boldsymbol{v}_1,\dots,\boldsymbol{v}_n\in\mathcal{S}$ we have $\mathcal{S}=\mathrm{Sp}\left(\boldsymbol{v}_1,\dots,\boldsymbol{v}_n\right)$
- Orthonormal vectors are LI

2.4. Gram-Schmidt

$\boldsymbol{u}_2=\boldsymbol{v}_2-\boldsymbol{u}_1\frac{\boldsymbol{u}_1^{\mathrm{H}}\boldsymbol{v}_2}{\boldsymbol{u}_1^{\mathrm{H}}\boldsymbol{u}_1}\dots$

2.5. Matrix Cookbook

$\mathrm{tr}(\boldsymbol{A}\boldsymbol{B})=\mathrm{tr}(\boldsymbol{B}\boldsymbol{A})$
$\mathrm{tr}(\boldsymbol{C}\boldsymbol{D}\boldsymbol{E})=\mathrm{tr}(\boldsymbol{E}\boldsymbol{C}\boldsymbol{D})=\mathrm{tr}(\boldsymbol{D}\boldsymbol{E}\boldsymbol{C})$
$(\boldsymbol{A}\boldsymbol{B})^{\mathrm{T}}=\boldsymbol{B}^{\mathrm{T}}\boldsymbol{A}^{\mathrm{T}}$
$(\boldsymbol{A}\boldsymbol{B})^{\mathrm{H}}=\boldsymbol{B}^{\mathrm{H}}\boldsymbol{A}^{\mathrm{H}}$
$(\boldsymbol{A}\boldsymbol{B})^{-1}=\boldsymbol{B}^{-1}\boldsymbol{A}^{-1}$
$\left(\boldsymbol{A}^{-1}\right)^{\mathrm{H}}=\left(\boldsymbol{A}^{\mathrm{H}}\right)^{-1}$

2.6. Determinant

$det:\mathbb{C}^{m\times m}\ni\boldsymbol{A}\mapsto\det\boldsymbol{A}\in\mathbb{C}$ having the following properties:

1. $\det\mathbf{I}_m=1$
2. If $\boldsymbol{A}$ has LD columns, then $\det\boldsymbol{A}=0$.
3. $\det\boldsymbol{A}$ is linear in the columns of $\boldsymbol{A}$.

For the determinant of MxM matrices the following is true:

1. For $m=1:\det A=A$.
2. $\det\boldsymbol{A}=\sum_{i=1}^m(-1)^{i+j}A_{i,j}\det\boldsymbol{A}^{[i,j]}$, for any $j\in\{1,\dots m\}$, where $\boldsymbol{A}^{[i,j]}$ is the matrix that results from $\boldsymbol{A}$ if the i-th row and the j-th column are removed.
3. $\det(\boldsymbol{A}\boldsymbol{B})=(\det\boldsymbol{A})(\det\boldsymbol{B})$
4. $\det\left(\boldsymbol{A}^{-1}\right)=(\det\boldsymbol{A})^{-1}$
5. $\det\left(\boldsymbol{A}^{\mathrm{T}}\right)=\det\boldsymbol{A}$

2.7. Eigenvalue Decomposition (EVD)

$$\boldsymbol{A}=\boldsymbol{B}\boldsymbol{\Lambda}\boldsymbol{B}^{-1}$$

$\mathrm{tr}\,\boldsymbol{A}=\sum_{i=1}^m\lambda_i$
$\det\boldsymbol{A}=\prod_{i=1}^m\lambda_i$
$\boldsymbol{A}^k=\boldsymbol{B}\boldsymbol{\Lambda}^k\boldsymbol{B}^{-1}$
Every scalar function which has a Taylor series expansion can be generalized to matrices:
$\boldsymbol{h}(\boldsymbol{A})=\boldsymbol{B}\boldsymbol{h}(\boldsymbol{\Lambda})\boldsymbol{B}^{-1}$

2.8. Hermitian Matrices

$$\boldsymbol{A}=\boldsymbol{A}^H\in\mathbb{C}^{m\times m}$$

Properties:

- m orthogonal (LI) eigenvectors $\rightarrow$ EVD exists
- real eigenvalues
- EVD has the form $\boldsymbol{A}=\boldsymbol{B}\boldsymbol{\Lambda}\boldsymbol{B}^{\mathrm{H}}$

2.9. Gramian Matrices

$$\exists\boldsymbol{C}\in\mathbb{C}^{m\times m}:\quad\boldsymbol{A}=\boldsymbol{C}\boldsymbol{C}^{\mathrm{H}}$$

Properties:

- Hermitian matrix
- non-negative real eigenvalues
- EVD exists

2.10. Sherman-Morrison-Woodbury identity

$$(\boldsymbol{A}+\boldsymbol{B}\boldsymbol{C}\boldsymbol{D})^{-1}=\boldsymbol{A}^{-1}-\boldsymbol{A}^{-1}\boldsymbol{B}\left(\boldsymbol{C}^{-1}+\boldsymbol{D}\boldsymbol{A}^{-1}\boldsymbol{B}\right)^{-1}\boldsymbol{D}\boldsymbol{A}^{-1}$$

$$\boldsymbol{A}\in\mathbb{C}^{M\times M},\boldsymbol{C}\in\mathbb{C}^{N\times N},\boldsymbol{B}\in\mathbb{C}^{M\times N},\boldsymbol{D}\in\mathbb{C}^{N\times M}$$
$$\mathrm{rank}\,\boldsymbol{A}=M,\mathrm{rank}\,\boldsymbol{C}=N$$

Tipps for Reformulation:

- insert $\boldsymbol{C}^{-1}\boldsymbol{C}$
- ausklammern wos geht

2.11. Singular Value Decomposition (SVD)

$$\boldsymbol{A}=\boldsymbol{U}\boldsymbol{\Sigma}\boldsymbol{V}^{\mathrm{H}}$$
$$\boldsymbol{U}\in\mathbb{C}^{m\times m},\quad\text{with}\quad\boldsymbol{U}^{-1}=\boldsymbol{U}^{\mathrm{H}}$$
$$\boldsymbol{V}\in\mathbb{C}^{n\times n},\quad\text{with}\quad\boldsymbol{V}^{-1}=\boldsymbol{V}^{\mathrm{H}}$$
$$\boldsymbol{\Sigma}\in\mathbb{R}^{m\times n},\quad\text{with}\quad\boldsymbol{\Sigma}_{i,j\neq i}=0,\quad\boldsymbol{\Sigma}_{i,i}\geq 0$$

- exists for every matrix

Relation to SVD:

- $\boldsymbol{V}$: SVD of $\boldsymbol{A}^{\mathrm{H}}\boldsymbol{A}$
- $\boldsymbol{U}$: SVD of $\boldsymbol{A}\boldsymbol{A}^{\mathrm{H}}$
- $\boldsymbol{\Sigma}\boldsymbol{\Sigma}^{\mathrm{T}}=\mathrm{diag}(\lambda_1\dots\lambda_m)$
- $s_i=\sqrt{\lambda_i},\quad 1\leq i\leq\min(m,n)$

$$\boldsymbol{A}=\boldsymbol{U}_1\boldsymbol{\Sigma}_1\boldsymbol{V}_1^{\mathrm{H}}=\begin{bmatrix}\boldsymbol{U}_1 & \boldsymbol{U}_2\end{bmatrix}\begin{bmatrix}\boldsymbol{\Sigma}_1 & \mathbf{O}\\ \mathbf{O} & \mathbf{O}\end{bmatrix}\begin{bmatrix}\boldsymbol{V}_1^{\mathrm{H}}\\ \boldsymbol{V}_2^{\mathrm{H}}\end{bmatrix}$$

Properties:

- $\boldsymbol{U}_1^{\mathrm{H}}\boldsymbol{U}_1=\boldsymbol{V}_1^{\mathrm{H}}\boldsymbol{V}_1=\mathbf{I}_r$
- $\boldsymbol{U}_2^{\mathrm{H}}\boldsymbol{U}_2=\mathbf{I}_{m-r},\ \mathbf{V}_2^{\mathrm{H}}\mathbf{V}_2=\mathbf{I}_{n-r}$
- $\boldsymbol{U}_1^{\mathrm{H}}\boldsymbol{U}_2=\mathbf{O}_{r,(m-r)},\ \boldsymbol{U}_2^{\mathrm{H}}\boldsymbol{U}_1=\mathbf{O}_{(m-r),r}$
- $\boldsymbol{V}_1^{\mathrm{H}}\boldsymbol{V}_2=\mathbf{O}_{r,(n-r)},\ \boldsymbol{V}_2^{\mathrm{H}}\boldsymbol{V}_1=\mathbf{O}_{(n-r),r}$
- $\boldsymbol{U}_1\boldsymbol{U}_1^{\mathrm{H}}+\boldsymbol{U}_2\boldsymbol{U}_2^{\mathrm{H}}=\mathbf{I}_m$
- $\boldsymbol{V}_1\boldsymbol{V}_1^{\mathrm{H}}+\boldsymbol{V}_2\boldsymbol{V}_2^{\mathrm{H}}=\mathbf{I}_n$
- $\mathrm{im}\,\boldsymbol{V}_1$ and $\mathrm{im}\,\boldsymbol{V}_2$ are complementary subspaces
- $\mathrm{im}\,\boldsymbol{U}_1$ and $\mathrm{im}\,\boldsymbol{U}_2$ are complementary subspaces

Four elementary subspaces of a matrix:
Image/column space:

- $\mathrm{im}\,\boldsymbol{A}=\mathrm{im}\,\boldsymbol{A}\boldsymbol{A}^{\mathrm{H}}=\mathrm{im}\,\boldsymbol{U}_1$
- $\dim(\mathrm{im}\,\boldsymbol{A})=\mathrm{rank}(\boldsymbol{A})=r$
- $\mathrm{rank}\,\boldsymbol{A}=\mathrm{rank}\,\boldsymbol{A}^{\mathrm{H}}=\mathrm{rank}\,\boldsymbol{A}^{\mathrm{T}}=\mathrm{rank}\,\boldsymbol{A}\boldsymbol{A}^{\mathrm{H}}=\mathrm{rank}\,\boldsymbol{A}^{\mathrm{H}}\boldsymbol{A}$
- $\boldsymbol{P}_{\mathrm{im}\boldsymbol{A}}=\boldsymbol{U}_1\boldsymbol{U}_1^{\mathrm{H}}$

Null space:

- $\text{null }\boldsymbol{A}=\mathrm{im}\,V_2$
- $\dim(\text{ null }\boldsymbol{A})=n-r$
- if $r=n$, the nullspace is $\{\boldsymbol{0}\}$
- $\boldsymbol{P}_{\text{null }\boldsymbol{A}}=\boldsymbol{V}_2\boldsymbol{V}_2^{\mathrm{H}}$

Left null space:
Right image space:

Complementary subspaces:
$\mathcal{S}_1\bigcap\mathcal{S}_2=\{\boldsymbol{0}\}\quad$ and $\quad\mathcal{S}_1\bigcup\mathcal{S}_2=\mathbb{C}^{m\times 1}$

2.12. Projectors

- $\mathrm{im}\,\boldsymbol{P}_{\mathcal{S}}=\mathcal{S}$
- $\boldsymbol{P}_{\mathcal{S}}=\boldsymbol{P}_{\mathcal{S}}^{\mathrm{H}}$
- $\boldsymbol{P}_{\mathcal{S}}\boldsymbol{P}_{\mathcal{S}}=\boldsymbol{P}_{\mathcal{S}}$

$$\boldsymbol{z}\in\mathcal{S}\iff\boldsymbol{P}_{\mathcal{S}}\boldsymbol{z}=\boldsymbol{z}$$

- Projectors are unique for their subspace

2.13. Eckart-Young

$\arg\min_{\boldsymbol{B}}\|\boldsymbol{A}-\boldsymbol{B}\|_{\mathrm{F}}^2,\quad$ s.t. $\quad\mathrm{rank}\,\boldsymbol{B}=k<r=\mathrm{rank}\,\boldsymbol{A}$

$$\boldsymbol{B}=\sum_{i=1}^k\boldsymbol{u}_i s_i\boldsymbol{v}_i^{\mathrm{H}}$$

$\|\boldsymbol{A}-\boldsymbol{B}\|_{\mathrm{F}}^2=\|\boldsymbol{\Sigma}-\boldsymbol{M}\|_{\mathrm{F}}^2$, where $\boldsymbol{A}=\boldsymbol{U}\boldsymbol{\Sigma}\boldsymbol{V}^{\mathrm{H}},\boldsymbol{M}=\boldsymbol{U}^{\mathrm{H}}\boldsymbol{B}\boldsymbol{V}$
$$=\sum_{i=1}|s_i-M_{i,i}|^2+\sum_{i>r}|M_{i,i}|^2+\sum_{i,j\neq i}|M_{i,j}|^2$$
Minimum: $M_{i,i}=0,\ i>r,\ M_{i,j\neq i}=0,\ M_{i,i}=s_i,\ i=1\dots k$

2.14. Frobenius Norm

$\|\boldsymbol{C}\|_{\mathrm{F}}^2 = \operatorname{tr}\left(\boldsymbol{C}^{\mathrm{H}}\boldsymbol{C}\right) = \operatorname{tr}\left(\boldsymbol{C}\boldsymbol{C}^{\mathrm{H}}\right)$

$\|\boldsymbol{C}\|_{\mathrm{F}}^2 = \left\|\boldsymbol{U}^{\mathrm{H}}\boldsymbol{C}\boldsymbol{V}\right\|_{\mathrm{F}}^2$

With $\boldsymbol{M} = \boldsymbol{U}^{\mathrm{H}}\boldsymbol{B}\boldsymbol{V}$:
$\|\boldsymbol{A} - \boldsymbol{B}\|_{\mathrm{F}}^2 = \|\boldsymbol{\Sigma} - \boldsymbol{M}\|_{\mathrm{F}}^2 = \sum_{i=1}^{r} |s_i - \boldsymbol{M}_{i,i}|^2 + \sum_{i>r} |\boldsymbol{M}_{i,i}|^2 + \sum_{i,j\neq i} |\boldsymbol{M}_{i,j}|^2$

2.15. Linear System of Equations

Exakt solution:
If and only if $b \in \operatorname{im} \boldsymbol{A}$, the system $\boldsymbol{A}\boldsymbol{w} = \boldsymbol{b}$, with $\boldsymbol{A} \in \mathbb{C}^{m\times n}$, has the following exact solution(s):

$$\boldsymbol{w} = \boldsymbol{V}_1\boldsymbol{\Sigma}_1^{-1}\boldsymbol{U}_1^{\mathrm{H}}\boldsymbol{b} + \boldsymbol{V}_2\boldsymbol{z}, \quad \text{for any} \quad \boldsymbol{z} \in \mathbb{C}^{(n-r)\times 1}, \quad r = \operatorname{rank} \boldsymbol{A}$$

The solution is unique iff. $\boldsymbol{A}$ has full column rank.

Minimum Norm Solution:
If $b \in \operatorname{im} \boldsymbol{A}$ and $\boldsymbol{A}$ has full row rank, the solution w_{MN}, of $\boldsymbol{A}w_{\mathrm{MN}} = b$, which has the smallest euclidian norm is given by:

$$\boldsymbol{w}_{\mathrm{MN}} = \boldsymbol{A}^{\mathrm{H}}\left(\boldsymbol{A}\boldsymbol{A}^{\mathrm{H}}\right)^{-1}\boldsymbol{b} \text{ (full row rank)}$$

Least Squares Solution:
While for $b \notin \operatorname{im} \boldsymbol{A}$, with $\boldsymbol{A} \in \mathbb{C}^{m\times n}$, there is no exact solution for the system $\boldsymbol{A}\boldsymbol{w} = \boldsymbol{b}$, an approximate solution, $\boldsymbol{w}_{\mathrm{LS}}$, can be defined as:

$$\boldsymbol{w}_{\mathrm{LS}} = \arg\min_{\boldsymbol{w}} \|\boldsymbol{A}\boldsymbol{w} - \boldsymbol{b}\|_2^2$$
$$\boldsymbol{w}_{\mathrm{LS}} = \left(\boldsymbol{A}^{\mathrm{H}}\boldsymbol{A}\right)^{-1}\boldsymbol{A}^{\mathrm{H}}\boldsymbol{b} \text{ (full column rank)}$$

2.16. Pseudoinverse

$$\boldsymbol{A}^+ = \begin{cases} \boldsymbol{V}_1\boldsymbol{\Sigma}_1^{-1}\boldsymbol{U}_1^{\mathrm{H}} & \text{for } \boldsymbol{A} \neq \boldsymbol{O} \\ \boldsymbol{A}^{\mathrm{T}} & \text{else.} \end{cases}$$

$$\boldsymbol{A}^+ = \begin{cases} \boldsymbol{A}^{\mathrm{H}}\left(\boldsymbol{A}\boldsymbol{A}^{\mathrm{H}}\right)^{-1} & \text{for full row-rank } \boldsymbol{A} \\ \left(\boldsymbol{A}^{\mathrm{H}}\boldsymbol{A}\right)^{-1}\boldsymbol{A}^{\mathrm{H}} & \text{for full column-rank } \boldsymbol{A} \\ \lim_{\epsilon\to 0} \boldsymbol{A}^{\mathrm{H}}\left(\boldsymbol{A}\boldsymbol{A}^{\mathrm{H}} + \epsilon\mathbf{I}\right)^{-1} & \\ \lim_{\epsilon\to 0} \left(\boldsymbol{A}^{\mathrm{H}}\boldsymbol{A} + \epsilon\mathbf{I}\right)^{-1}\boldsymbol{A}^{\mathrm{H}} & \end{cases}$$

Relation to the projectors:

- $\boldsymbol{P}_{\operatorname{im}\boldsymbol{A}} = \boldsymbol{A}\boldsymbol{A}^+ = \boldsymbol{U}_1\boldsymbol{U}_1^{\mathrm{H}}$
- $\boldsymbol{P}_{\text{null } \boldsymbol{A}} = \boldsymbol{I} - \boldsymbol{A}^+\boldsymbol{A} = \boldsymbol{V}_2\boldsymbol{V}_2^{\mathrm{H}}$

3. Random Processes

3.1. Discrete random processes

Definition: function $x[n]$ which is selected from an ensemble of possible functions by random.

Properties obtained by averaging over the ensemble:

Expectation function:
$$\mu_x[n] = \mathrm{E}[x[n]]$$
Autocorrelation function:
$$r_x[n,k] = \mathrm{E}[x[n]x[n-k]]$$
Autocovariance function:
$$c_x[n,k] = \mathrm{E}\left[(x[n] - \mu_x[n])\,(x[n-k] - \mu_x[n-k])\right]$$
$$= r_x[n,k] - \mu_x[n]\mu_x[n-k]$$
Cross-correlation function:
$$r_{x,y}[n,k] = \mathrm{E}[x[n]y[n-k]]$$
Cross-covariance function:
$$c_{x,y}[n,k] = \mathrm{E}\left[(x[n] - \mu_x[n])\,(y[n-k] - \mu_y[n-k])\right]$$
$$= r_{x,y}[n,k] - \mu_x[n]\mu_y[n-k]$$

3.2. Approximations obtained by averaging over time:

Expectation function:
$$\hat{\mu}_x^{[N]}[n] = \frac{1}{2N+1}\sum_{i=-N}^{N} x[n+i]$$
Cross-correlation function:
$$\hat{r}_{x,y}^{[N]}[n,k] = \frac{1}{2N+1}\sum_{i=-N}^{N} x[n+i]y[n+i-k]$$

The approximations are random processes themselves.

3.3. Wide-sense-stationary (WSS) processes

$\mu_x[n]$, $r_x[n,k]$, $c_x[n,k]$, $r_{x,y}[n,k]$ and $c_{x,y}[n,k]$ are independent of the time index n.

$$\mu_x = \mathrm{E}[x[n]]$$
$$r_x[k] = \mathrm{E}[x[n]x[n-k]],\ r_x[k] = r_x[-k]$$
$$r_{x,y}[k] = \mathrm{E}[x[n]y[n-k]],\ r_{x,y}[k] = r_{y,x}[-k]$$
$$\mathrm{E}\left[\hat{\mu}_x^{[N]}[n]\right] = \mu_x, \quad \mathrm{E}\left[\hat{r}_x^{[N]}[n,k]\right] = r_x[k]$$

For zero-mean processes: $\mu_x = 0$, $\mu_y = 0$, $\ldots$: correlation and covariance are identical

3.4. Ergodic processes

Averages over ensembles can be purely obtained from averages over time. Ergodicity implies WSS.

$$\lim_{N\to\infty} \hat{\mu}_x^{[N]}[n] = \mu_x$$
$$\lim_{N\to\infty} \hat{r}_x^{[N]}[n,k] = r_x[k]$$

3.5. Complex processes

$u[n] = x[n] + \mathrm{j}y[n]$

Complex autocorrelation function:
$r_u[k] = \mathrm{E}\left[u[n]u^*[n-k]\right]$
$r_u[k] = r_x[k] + r_y[k] + \mathrm{j}\left(r_{x,y}[-k] - r_{x,y}[k]\right)$

Adjunct complex autocorrelation function:
$\tilde{r}_u[k] = \mathrm{E}[u[n]u[n-k]]$
$\tilde{r}_u[k] = r_x[k] - r_y[k] + \mathrm{j}\left(r_{x,y}[k] + r_{x,y}[-k]\right)$

Reformulation:
$r_x[k] = \frac{1}{2}\operatorname{Re}\{r_u[k] + \tilde{r}_u[k]\}$
$r_y[k] = \frac{1}{2}\operatorname{Re}\{r_u[k] - \tilde{r}_u[k]\}$
$r_{x,y}[k] = -\frac{1}{2}\operatorname{Im}\{r_u[k] - \tilde{r}_u[k]\}$

3.6. Proper WSS processes

Equivalent definitions (iff):
$\forall k: \quad \tilde{r}_u[k] = 0$
$\forall k: \quad r_x[k] = r_y[k], \quad$ and $\quad \forall k: \quad r_{x,y}[k] = -r_{x,y}[-k]$

$$\mathrm{E}\left[\boldsymbol{u}[n]\boldsymbol{u}^{\mathrm{T}}[n]\right] = \begin{bmatrix} \tilde{r}_u[0] & \tilde{r}_u[1] & \tilde{r}_u[2] & \cdots \\ \tilde{r}_u[1] & \tilde{r}_u[0] & \tilde{r}_u[1] & \cdots \\ \vdots & \ddots & \ddots & \vdots \end{bmatrix} = \boldsymbol{O}$$

$\Rightarrow$ The autocorrelation function is completely described by the complex autocorrelation function.

4. Overview

4.1. Linear estimation for matrices

Problem: Find $\hat{\boldsymbol{S}}$ from $\boldsymbol{X}$

$$\boldsymbol{X} = \boldsymbol{A}\boldsymbol{S} + \boldsymbol{\Upsilon}$$

Four different cases:

1. nothing is known (except structure) $\to$ MUSIC
2. only $\boldsymbol{A}$ is known $\to$ Least Squares
3. $\boldsymbol{A}$ and $\mathrm{E}\left[\boldsymbol{\Upsilon}\boldsymbol{\Upsilon}^{\mathrm{H}}\right]$ are known $\to$ BLUE
4. $\boldsymbol{A}$, $E\left[\boldsymbol{\Upsilon}\boldsymbol{\Upsilon}^{\mathrm{H}}\right]$ and $\mathrm{E}\left[\boldsymbol{S}\boldsymbol{S}^{\mathrm{H}}\right]$ are known

5. Kolmogorov-Wiener Filters

5.1. Linear Filters

$$u = h * s$$

If h has $K+1$ coefficients, its memory is K.
If M output samples are required: $\boldsymbol{H} \in \mathbb{C}^{M\times(M+K)}$

5.2. Kolmogorov-Wiener filter for SISO/ time domain equalizer for the linear multipath channel

Optimization problem:

$$\mathbf{MSE} = \mathrm{E}[|\underbrace{y[n] - d[n]}_{e[n]}|^2]$$
$$(w[0],\ldots,w[M-1])_{\mathrm{opt}} = \arg\min_{(w[0],\ldots,w[M-1])} \mathrm{MSE}$$

$\boldsymbol{u}[n] \in \mathbb{C}^{M\times 1}$, $\boldsymbol{s}[n] \in \mathbb{C}^{N\times 1}$, $\boldsymbol{H} \in \mathbb{C}^{M\times N}$, $N = M + K$

$$y[n] = \boldsymbol{w}^{\mathrm{H}}\boldsymbol{u}[n] = \boldsymbol{w}^{\mathrm{H}}(\boldsymbol{H}\boldsymbol{s}[n] + \boldsymbol{v}[n])$$

$\boldsymbol{R} = \mathrm{E}\left[\boldsymbol{u}[n]\boldsymbol{u}^{\mathrm{H}}[n]\right] = \boldsymbol{H}\boldsymbol{R}_s\boldsymbol{H}^{\mathrm{H}} + \boldsymbol{H}\boldsymbol{R}_{s,v} + \boldsymbol{R}_{v,s}\boldsymbol{H}^{\mathrm{H}} + \boldsymbol{R}_v$
$\boldsymbol{p} = \mathrm{E}\left[\boldsymbol{u}[n]d^*[n]\right]$
$\sigma_s^2 = \mathrm{E}\left[d[n]d^*[n]\right]$

Solutions in terms of u:

$$\mathrm{E}\left[\boldsymbol{u}[n]e^*[n]\right] = 0 \text{ (principle of orthogonality)}$$
$$\boldsymbol{w}_{\mathrm{opt}} = \boldsymbol{R}^{-1}\boldsymbol{p}, \text{ or: } \boldsymbol{w}_{\mathrm{opt}}^{\mathrm{H}} = \boldsymbol{p}^{\mathrm{H}}\boldsymbol{R}^{-1}$$
$$\mathrm{MSE}_{\min} = \sigma_s^2 - \boldsymbol{p}^{\mathrm{H}}\boldsymbol{R}^{-1}\boldsymbol{p}$$

$\boldsymbol{w}_{\mathrm{opt}}$ minimizes MSE if: $\boldsymbol{R} > 0 \Leftrightarrow \boldsymbol{R}$ invertible ($\boldsymbol{R}$ is Gramian)
In general: $\mathbf{MSE} = \boldsymbol{w}^{\mathrm{H}}\boldsymbol{R}\boldsymbol{w} - \boldsymbol{w}^{\mathrm{H}}\boldsymbol{p} - \boldsymbol{p}^{\mathrm{H}}\boldsymbol{w} + \sigma_s^2$

Noise variance: $E[(\boldsymbol{w}_{\mathrm{opt}}^H\boldsymbol{v}[n])(\boldsymbol{w}_{\mathrm{opt}}^H\boldsymbol{v}[n])^H]$
Overall impulse response: coefficients given by $\boldsymbol{w}_{\mathrm{opt}}^H\boldsymbol{H}$

Special case: $d[n] = s[n-l]$, s and v are uncorrelated

$$\boldsymbol{w}_{\mathrm{opt}} = \left(\boldsymbol{H}\boldsymbol{R}_{\boldsymbol{s}}\boldsymbol{H}^{\mathrm{H}} + \boldsymbol{R}_{\boldsymbol{v}}\right)^{-1}\boldsymbol{H}\boldsymbol{R}_{\boldsymbol{s}}\mathbf{e}_{l+1}$$
$$= \boldsymbol{R}_{\boldsymbol{v}}^{-1}\boldsymbol{H}\left(\boldsymbol{R}_{\boldsymbol{s}}^{-1} + \boldsymbol{H}^{\mathrm{H}}\boldsymbol{R}_{\boldsymbol{v}}^{-1}\boldsymbol{H}\right)^{-1}\mathbf{e}_{l+1}$$

$\boldsymbol{R} = \boldsymbol{H}\boldsymbol{R}_s\boldsymbol{H}^{\mathrm{H}} + \boldsymbol{R}_v$
$\boldsymbol{p} = \boldsymbol{H}\boldsymbol{R}_s\mathbf{e}_{l+1}$
$\boldsymbol{R}_s = \mathrm{E}\left[\boldsymbol{s}[n]\boldsymbol{s}^{\mathrm{H}}[n]\right]$
$\boldsymbol{R}_{s,v} = \mathrm{E}\left[\boldsymbol{s}[n]\boldsymbol{v}^{\mathrm{H}}[n]\right] = \boldsymbol{O}$
$\boldsymbol{R}_v = \mathrm{E}\left[\boldsymbol{v}[\boldsymbol{n}]\boldsymbol{v}^{\mathrm{H}}[\boldsymbol{n}]\right]$

Optimization of l_{opt}:

$$l_{\mathrm{opt}} = \arg\min_{l\in\{0,1,\ldots N-1\}} \sigma_s^2 - \boldsymbol{p}^{\mathrm{H}}\boldsymbol{R}^{-1}\boldsymbol{p}$$
$$= \arg\max_{l\in\{0,1,\ldots N-1\}} \boldsymbol{p}^{\mathrm{H}}\boldsymbol{R}^{-1}\boldsymbol{p}$$
$$= \text{argmin}\sigma_s^2 - \mathbf{e}_{l+1}^{\mathrm{T}}\boldsymbol{R}_{\boldsymbol{s}}\boldsymbol{H}^{\mathrm{H}}\left(\boldsymbol{H}\boldsymbol{R}_{\boldsymbol{s}}\boldsymbol{H}^{\mathrm{H}} + \boldsymbol{R}_v\right)^{-1}\boldsymbol{H}\boldsymbol{R}_{\boldsymbol{s}}\mathbf{e}_{l+1}$$

This is done by trying a few neighbors of $N/2 = (M+K)/2$.

Noise variance: $\sigma_n^2 = E[(\boldsymbol{w}_{\mathrm{opt}}^H\boldsymbol{v}[n])(\boldsymbol{w}_{\mathrm{opt}}^H\boldsymbol{v}[n])^H]$
Overall impulse response: coefficients given by $\boldsymbol{h} \leftarrow \boldsymbol{w}_{\mathrm{opt}}^H\boldsymbol{H}$
Signal-to-interference ratio:

$$\text{SINR} = \frac{\sigma_s'^2|h[l]|^2}{\sigma_s'^2\sum_{i,i\neq l}|h[i]|^2 + \sigma_n^2}, \quad \text{assuming } \boldsymbol{R}_s = \sigma_s^2\boldsymbol{I}$$

.

5.3. Kolmogorov-Wiener filter for diversity reception

L multipath channels with just **one transmit signal**:
$\boldsymbol{u}_i[n] = \boldsymbol{H}_i\boldsymbol{s}[n] + \boldsymbol{v}_i[n], i \in \{1,2,\ldots L\}$, $\boldsymbol{H}_i \in \mathbb{C}^{M\times(M+K)}$

$$\underbrace{\begin{bmatrix} u_1[n] \\ u_2[n] \\ \vdots \\ u_L[n] \end{bmatrix}}_{\boldsymbol{u}[n]} = \underbrace{\begin{bmatrix} H_1 \\ H_2 \\ \vdots \\ H_L \end{bmatrix}}_{\boldsymbol{H}\in\mathbb{C}^{LM\times(M+K)}} s[n] + \underbrace{\begin{bmatrix} v_1[n] \\ v_2[n] \\ \vdots \\ v_L[n] \end{bmatrix}}_{\boldsymbol{v}[n]}$$

Same formula as for SISO but with different matrices:
$\boldsymbol{w}_{\mathrm{opt}} = \left(\boldsymbol{H}\boldsymbol{R}_{\boldsymbol{s}}\boldsymbol{H}^{\mathrm{H}} + \boldsymbol{R}_{\boldsymbol{v}}\right)^{-1}\boldsymbol{H}\boldsymbol{R}_{\boldsymbol{s}}\mathbf{e}_{l+1}$
$\boldsymbol{R}_{\boldsymbol{s}} = \mathrm{E}\left[\boldsymbol{s}[n]\boldsymbol{s}[n]^{\mathrm{H}}\right] \in \mathbb{C}^{(M+K)\times(M+K)}$
$\boldsymbol{R}_{\boldsymbol{v}} = \mathrm{E}\left[\boldsymbol{v}[n]\boldsymbol{v}[n]^{\mathrm{H}}\right] \in \mathbb{C}^{LM\times LM}$
$y[n] = \boldsymbol{w}_{\mathrm{opt}}^{\mathrm{H}}\boldsymbol{u}[n]$
usually $l_{\mathrm{opt}} = \lfloor(M+K)/2\rfloor$

Diversity reception

- one signal is transmitted over multiple channels
- others can still be used if one breaks (redundancy)
- space-diversity, frequency-diversity, time-diversity, or combinations

5.4. Kolmogorov-Wiener filter for multi-streaming

L channels with **different transmit signals**:

$$\underbrace{\begin{bmatrix} \boldsymbol{u}_1[n] \\ \boldsymbol{u}_2[n] \\ \vdots \\ \boldsymbol{u}_{m_{\mathrm{Rx}}}[n] \end{bmatrix}}_{\boldsymbol{u}[n]} = \underbrace{\begin{bmatrix} \boldsymbol{H}_{1,1} & \boldsymbol{H}_{1,2} & \cdots & \boldsymbol{H}_{1,m_{\mathrm{Tx}}} \\ \boldsymbol{H}_{2,1} & \boldsymbol{H}_{2,2} & \cdots & \boldsymbol{H}_{2,m_{\mathrm{Tx}}} \\ \vdots & \vdots & \vdots & \vdots \\ \boldsymbol{H}_{m_{\mathrm{Rx}},1} & \boldsymbol{H}_{m_{\mathrm{Rx}},2} & \cdots & \boldsymbol{H}_{m_{\mathrm{Rx}},m_{\mathrm{Tx}}} \end{bmatrix}}_{\boldsymbol{H}} \underbrace{\begin{bmatrix} \boldsymbol{s}_1[n] \\ \boldsymbol{s}_2[n] \\ \vdots \\ \boldsymbol{s}_{m_{\mathrm{Tx}}}[n] \end{bmatrix}}_{\boldsymbol{s}[n]} + \underbrace{\begin{bmatrix} \boldsymbol{v}_1[n] \\ \boldsymbol{v}_2[n] \\ \vdots \\ \boldsymbol{v}_{m_{\mathrm{Rx}}}[n] \end{bmatrix}}_{\boldsymbol{v}[n]}$$

$\boldsymbol{H}_{i,j} \in \mathbb{C}^{M\times(M+K)}$ connects j-th transmitter to i-th receiver
$\mathbf{H} \in \mathbb{C}^{Mm_{\mathrm{Rx}}\times(M+K)m_{\mathrm{Tx}}}$
$\boldsymbol{s}[n] \in \mathbb{C}^{(M+K)m_{\mathrm{Tx}}}$
$\boldsymbol{v}[n] \in \mathbb{C}^{Mm_{\mathrm{Rx}}}$
K is the maximum channel memory of all channels

Main difference to before: Ordering of the elements in $\boldsymbol{s}[n]$ has changed to groups of time series.

Solution:

$$\boldsymbol{w}_{j,\mathrm{opt}} = \underbrace{\left(\boldsymbol{H}\boldsymbol{R}_{\boldsymbol{s}}\boldsymbol{H}^{\mathrm{H}} + \boldsymbol{R}_v\right)^{-1}}_{\boldsymbol{R}^{-1}} \underbrace{\boldsymbol{H}\boldsymbol{R}_{\boldsymbol{s}}\mathbf{e}_{(M+K)(j-1)+l_{j+1}}}_{\boldsymbol{p}}$$
$$y_j[n] = \boldsymbol{w}_{j,\mathrm{opt}}^{\mathrm{H}}\boldsymbol{u}[n], \quad j \in \{1,2,\ldots,m_{\mathrm{Tx}}\}$$

$\boldsymbol{w}_{j,\mathrm{opt}}$ is the optimum filter for recovering the signal of the j-th transmitter with time delay $l_j \in \{0,\ldots,M+K-1\}$

5.5. Kolmogorov-Wiener Filter for SISO

Setup:

$$\boldsymbol{u} = \boldsymbol{h}\boldsymbol{s} + \boldsymbol{v}, \quad \boldsymbol{s} = \boldsymbol{w}^H\boldsymbol{u}$$
$$\boldsymbol{R}_{\boldsymbol{s}} = 1, \quad \boldsymbol{R}_v = \mathbf{I}, \quad s, v \text{ uncorrelated}$$

MSE solution:

$$\boldsymbol{w} = (\boldsymbol{I} + \boldsymbol{h}\boldsymbol{h}^H)^{-1}\boldsymbol{h} = \frac{\boldsymbol{h}}{1 + \boldsymbol{h}^H\boldsymbol{h}}$$

5.6. General Properties of Kolmogorov-Wiener

- Linear filter: can only use first order correlations of d and u, $\to p$ must be $\neq 0$
- Filtering without noise: MSE drops exponentially when increasing reconstruction filter length (if $\boldsymbol{H}$ has full column rank)
- Filtering with noise: MSE saturates and cannot drop exponentially

5.7. Kolmogorov-Wiener Filter for SNR $\to \infty$

Setup:

$$\boldsymbol{u} = \boldsymbol{H}\boldsymbol{s} + \boldsymbol{v}, \quad \hat{\boldsymbol{s}} = \boldsymbol{w}^H \boldsymbol{u}$$

$$\boldsymbol{R_s} = \sigma_s^2 \mathbf{I}, \quad \boldsymbol{R}_v = \sigma_v^2 \mathbf{I}, \quad s, v \text{ uncorrelated}$$

MSE solution:

$$\lim_{\sigma_s^2/\sigma_v^2 \to \infty} \boldsymbol{w}_{\text{opt}}^{\mathrm{H}} = \begin{cases} \mathbf{e}_{l+1}^{\mathrm{T}} \boldsymbol{H}^{\mathrm{H}} \left(\boldsymbol{H}\boldsymbol{H}^{\mathrm{H}}\right)^{-1}, & \text{full row rank } \boldsymbol{H} \\ \mathbf{e}_{l+1}^{\mathrm{T}} \left(\boldsymbol{H}^{\mathrm{H}}\boldsymbol{H}\right)^{-1} \boldsymbol{H}^{\mathrm{H}}, & \text{full column rank} \boldsymbol{H} \end{cases}$$

$$= \mathbf{e}_{l+1}^{\mathrm{T}} \boldsymbol{H}^{+}$$

Perfect reconstruction only if H has full column rank:

$$\hat{\boldsymbol{s}}_{l+1} = \mathbf{e}_{l+1}^{\mathrm{T}} \boldsymbol{H}^{+} \boldsymbol{H} \boldsymbol{s} = \mathbf{e}_{l+1}^{\mathrm{T}} \underbrace{\left(\boldsymbol{H}^{\mathrm{H}}\boldsymbol{H}\right)^{-1} \boldsymbol{H}^{\mathrm{H}}\boldsymbol{H}}_{\mathbf{I}} \boldsymbol{s} = \boldsymbol{s}_{l+1}$$

Imperfect reconstruction if H has not full column rank:

$$\hat{s}_{l+1} = \mathbf{e}_{l+1}^{\mathrm{T}} \underbrace{\boldsymbol{H}^{+}\boldsymbol{H}}_{\neq \mathbf{I}} \boldsymbol{s} \neq \boldsymbol{s}_{l+}$$

Because with $\operatorname{rank} \boldsymbol{H} < N$:

$$\boldsymbol{H}^{+}\boldsymbol{H} = \mathbf{I} - \boldsymbol{P}_{\text{null } \boldsymbol{H}}$$

$$\dim \text{ null } \boldsymbol{H} = N - \operatorname{rank} \boldsymbol{H} > 0$$

$$\boldsymbol{P}_{\text{null } \boldsymbol{H}} \neq \mathbf{O}$$

5.8. Steepest Descent Algorithm (SDA)

Kolmogorov-Wiener filters need to solve $\boldsymbol{R}\boldsymbol{w}_{\text{opt}} = \boldsymbol{p}$, where:

$\boldsymbol{R} = \mathrm{E}\left[\boldsymbol{u}[n]\boldsymbol{u}^{\mathrm{H}}[n]\right]$ and $\boldsymbol{p} = \mathrm{E}\left[\boldsymbol{u}[n]d^*[n]\right]$

Problems with solving explicitly

- Accuracy: computation of m^2 complex numbers for R^{-1} and matrix vector multiplication add errors when m is large
- Computational load: Recomputing R^{-1} at every time step is costly
- Gaussian elimination needs to restart the entire computation every time
- Triangular schemes bring no benefit as R also changes

Gradient of MSE:

$\mathrm{MSE}(w) = F\left(\frac{1}{2}\left(w + w^{\star}\right), \frac{1}{2}\left(w - w^{\star}\right)/\mathrm{j}\right) = G\left(w, w^{*}\right)$

$$\mathrm{dMSE} = \left(\frac{\partial G}{\partial \boldsymbol{w}}\right)^{\mathrm{T}} \mathrm{d}\boldsymbol{w} + \left(\frac{\partial G}{\partial \boldsymbol{w}^{*}}\right)^{\mathrm{T}} \mathrm{d}\boldsymbol{w}^{*}$$

$$= \left(\frac{\partial G^{*}}{\partial \boldsymbol{w}^{*}}\right)^{\mathrm{H}} \mathrm{d}\boldsymbol{w} + \left(\left(\frac{\partial G}{\partial \boldsymbol{w}^{*}}\right)^{\mathrm{H}} \mathrm{d}\boldsymbol{w}\right)^{*}$$

$$= 2\operatorname{Re}\left\{\left(\frac{\partial G}{\partial w^{\star}}\right)^{\mathrm{H}} \mathrm{d}w\right\} = 2\operatorname{Re}\left\{(\boldsymbol{R}w - \boldsymbol{p})^{\mathrm{H}} \mathrm{d}\boldsymbol{w}\right\}$$

$$\leq 2\left|(\boldsymbol{R}w - \boldsymbol{p})^{\mathrm{H}} \mathrm{d}\boldsymbol{w}\right|, \text{ with equality for } \mathrm{d}\boldsymbol{w} = (\boldsymbol{R}\boldsymbol{w} - \boldsymbol{p})\mathrm{d}t$$

Gradient descent:

$$\boldsymbol{w}_{n+1} = \boldsymbol{w}_n - (\boldsymbol{R}\boldsymbol{w}_n - \boldsymbol{p})\,\mu, \quad \mu > 0$$

$$\boldsymbol{w}_{n+1} = (\mathbf{I} - \mu\boldsymbol{R})\boldsymbol{w}_n + \mu\boldsymbol{p}$$

Convergence:

$$\boldsymbol{c}_n = \boldsymbol{w}_n - \boldsymbol{w}_{\text{opt}}, \boldsymbol{c}_{n+1} = (\mathbf{I} - \mu\boldsymbol{R})\boldsymbol{c}_n$$

$$\boldsymbol{R} = \boldsymbol{Q}\boldsymbol{\Lambda}\boldsymbol{Q}^{H}, \boldsymbol{z}_n = \mathbf{Q}^{\mathrm{H}}\boldsymbol{c}_n, \boldsymbol{z}_{n+1} = (\mathbf{I} - \mu\boldsymbol{\Lambda})\boldsymbol{z}_n$$

$$\Rightarrow \forall i \in \{1, 2, \ldots, m\} : |1 - \mu\lambda_i| < 1 \Leftrightarrow 0 < \mu < 2/\lambda_{\max}$$

sufficient (not necessary): $0 < \mu < \frac{2}{\operatorname{tr} R}$

Steepest Descent Algorithm:

μ_{opt} is obtained by minimizing MSE_{n+1} wrt. μ, i.e. $\frac{\partial \mathrm{MSE}_{n+1}}{\partial \mu^{*}} = 0$

$\mathbf{MSE}_n = \boldsymbol{w}_n^{\mathrm{H}}\boldsymbol{R}\boldsymbol{w}_n - \boldsymbol{w}_n^{\mathrm{H}}\boldsymbol{p} - \boldsymbol{p}^{\mathrm{H}}\boldsymbol{w}_n + \sigma_d^2$

$\boldsymbol{w}_{n+1} = \boldsymbol{w}_n + \mu\boldsymbol{r}_n, \quad \boldsymbol{r}_n = \boldsymbol{p} - \boldsymbol{R}\boldsymbol{w}_n$

STEEPEST DESCENT ALGORITHM (SDA):

Input: $\mathbb{C}^{m\times m} \ni \boldsymbol{R} = \boldsymbol{R}^{\mathrm{H}} > \mathbf{0}$, and $\boldsymbol{p} \in \mathbb{C}^{m\times 1}$, some $\epsilon > 0$ and initial value $\boldsymbol{w}_0$

Output: An approximate solution of $\boldsymbol{R}\boldsymbol{w} = \boldsymbol{p}$ for $\boldsymbol{w}$, such that $\|\boldsymbol{R}\boldsymbol{w} - \boldsymbol{p}\|_2^2 \leq \epsilon$.

1. Init: $\boldsymbol{w} \leftarrow \boldsymbol{w}_0$
2. Search direction: $\boldsymbol{r} \leftarrow \boldsymbol{p} - \boldsymbol{R}\boldsymbol{w}$
3. Test: if $\boldsymbol{r}^{\mathrm{H}}\boldsymbol{r} \leq \epsilon$ terminate and return $\boldsymbol{w}$.
4. Step-size: $\mu \leftarrow \boldsymbol{r}^{\mathrm{H}}\boldsymbol{r}/(\boldsymbol{r}^{\mathrm{H}}\boldsymbol{R}\boldsymbol{r})$
5. Update: $\boldsymbol{w} \leftarrow \boldsymbol{w} + \mu\boldsymbol{r}$
6. Continue: goto step 2.

Complexity: $M^2 \log M$ scalar arithmetic operations/series time-steps

Complexity with full parallelization: $(\log M)^2$ time-steps

Steps 2-5 require $4m^2 + 5n - 1$ scalar arithmetic operations per iteration

Number of iterations: $\approx 15(\log(M) - 1), M > 15$

5.9. SDA for constrained optimization

Problem:

$$\min_{\boldsymbol{w}} \boldsymbol{w}^{\mathrm{H}}\boldsymbol{A}\boldsymbol{w}, \quad \text{s.t.} \quad \boldsymbol{B}^{\mathrm{H}}\boldsymbol{w} = \boldsymbol{c}$$

$\mathbb{C}^{M\times M} \ni \boldsymbol{A} = \boldsymbol{A}^{\mathrm{H}} > \mathbf{0}, \boldsymbol{B} \in \mathbb{C}^{M\times L}, \boldsymbol{c} \in \mathbb{C}^{L\times 1}$, rank $\boldsymbol{B} = L$

Ansatz:

Decompose solution into fixed term $\boldsymbol{w}_q$ and variable term $\boldsymbol{z}$

$$\boldsymbol{B}^{\mathrm{H}}\boldsymbol{w}_{\mathrm{q}} = \boldsymbol{c}, \quad \boldsymbol{z} \in \text{ null } \boldsymbol{B}^{\mathrm{H}}$$

$$\boldsymbol{w} = \boldsymbol{w}_q - \boldsymbol{z}$$

Parameterize $\boldsymbol{z}$ by $\boldsymbol{w}_a$:

$$\boldsymbol{B} = \underbrace{\begin{bmatrix} \boldsymbol{U}_1 & \boldsymbol{U}_2 \end{bmatrix}}_{\boldsymbol{U}} \begin{bmatrix} \boldsymbol{\Sigma}_1 & \mathbf{O} \\ \mathbf{O} & \mathbf{O} \end{bmatrix} \begin{bmatrix} \boldsymbol{V}_1^{\mathrm{H}} \\ \boldsymbol{V}_2^{\mathrm{H}} \end{bmatrix}$$

$$\boldsymbol{z} = \boldsymbol{U}_2\boldsymbol{w}_{\mathrm{a}}, \quad \boldsymbol{w}_{\mathrm{a}} \in \mathbb{C}(M - L) \times 1 \Rightarrow \boldsymbol{z} \in \operatorname{null}\boldsymbol{B}^{H} \forall \boldsymbol{w}_a$$

Obtaining $\boldsymbol{U}_2$ without SVD:

1. Init: $\boldsymbol{U} \leftarrow \begin{bmatrix} \boldsymbol{B} & \boldsymbol{F} \end{bmatrix}$, where $\boldsymbol{F} \in \mathbb{C}^{M\times(M-L)}$ has i.i.d. random components.
2. Orthogonalize with all yet orthogonalized columns. For $i = 0$ to $M - 1$:
   $$\boldsymbol{u}_i \leftarrow \boldsymbol{u}_i - \sum_{j=1}^{i-1} \boldsymbol{u}_j \left(\boldsymbol{u}_j^{\mathrm{H}}\boldsymbol{u}_i\right) / \left(\boldsymbol{u}_j^{\mathrm{H}}\boldsymbol{u}_j\right)$$
3. Normalize: For $i \in \{L+1, L+2, \ldots M\}$ do $\boldsymbol{u}_i \leftarrow \boldsymbol{u}_i / \sqrt{\boldsymbol{u}_i^{\mathrm{H}}\boldsymbol{u}_i}$
4. Output: $\boldsymbol{U}_2 \leftarrow \left[\boldsymbol{u}_{L+1}\boldsymbol{u}_{L+2} \cdots \boldsymbol{u}_M\right] \in \mathbb{C}^{M\times(M-L)}$

Finding w_q by SDA:

$$w_q = \boldsymbol{B}\underbrace{\left(\boldsymbol{B}^{\mathrm{H}}\boldsymbol{B}\right)^{-1} c}_{q}$$

$$\left(\boldsymbol{B}^{\mathrm{H}}\boldsymbol{B}\right) q = c$$

$$w_q = \boldsymbol{B}q$$

Reformulate:

$$\min_{\boldsymbol{w}_{\mathrm{a}}} \left(\boldsymbol{w}_{\mathrm{q}}^{\mathrm{H}} - \boldsymbol{w}_{\mathrm{a}}^{\mathrm{H}}\boldsymbol{U}_2^{\mathrm{H}}\right) \boldsymbol{A} \left(\boldsymbol{w}_{\mathrm{q}} - \boldsymbol{U}_2\boldsymbol{w}_{\mathrm{a}}\right)$$

Solution: run SDA with:

$$\boldsymbol{R} = \boldsymbol{U}_2^{\mathrm{H}}\boldsymbol{A}\boldsymbol{U}_2 \in \mathbb{C}^{(M-L)\times(M-L)}, \quad \text{and} \quad \boldsymbol{p} = \boldsymbol{U}_2^{\mathrm{H}}\boldsymbol{A}\boldsymbol{w}_{\mathrm{q}}$$

5.10. Steepest Descent Procedure (SDP)

Problem: R and p are unknown and need to be estimated.

Drop assumption that $u[n]$ is WSS (e.g. channel changes) $\to \boldsymbol{R}[n]$

Estimation with exponential weighting:

$$\widehat{\boldsymbol{R}}[n] = \frac{\sum_{k=0}^{\infty} \boldsymbol{u}[n-k]\boldsymbol{u}^{\mathrm{H}}[n-k]\alpha[k]}{\sum_{k=0}^{\infty} \alpha[k]}$$

$$\alpha[k] = \begin{cases} 1 & \text{for } k = 0 \\ \eta^k & \text{for } k > 0 \\ 0 & \text{else} \end{cases}, \quad \sum_{k=0}^{\infty} \eta^k = \frac{1}{1-\eta}$$

$$\widehat{\boldsymbol{R}}[n] = \eta\widehat{\boldsymbol{R}}[n-1] + (1-\eta)\boldsymbol{u}[n]\boldsymbol{u}^{\mathrm{H}}[n]$$

$$\widehat{\boldsymbol{p}}[n] = \eta\widehat{\boldsymbol{p}}[n-1] + (1-\eta)\boldsymbol{u}[n]d^{*}[n]$$

STEEPEST DESCENT PROCEDURE (SDP)

1. Init: $\boldsymbol{w} \leftarrow \mathbf{0}_M$, $\widehat{\boldsymbol{R}} \leftarrow \mathbf{O}_{M\times M}$, $\widehat{\boldsymbol{p}} \leftarrow \mathbf{0}_M$, $n \leftarrow 0$
2. Update auto-correlation: $\widehat{\boldsymbol{R}} \leftarrow \eta\widehat{\boldsymbol{R}} + (1-\eta)\boldsymbol{u}[n]\boldsymbol{u}^{\mathrm{H}}[n]$
3. Update cross-correlation: If $d[n]$ available: $\widehat{\boldsymbol{p}} \leftarrow \eta\widehat{\boldsymbol{p}} + (1-\eta)\boldsymbol{u}[n]d^{*}[n]$
4. Update weight: $\boldsymbol{r} \leftarrow \widehat{\boldsymbol{p}} - \widehat{\boldsymbol{R}}\boldsymbol{w}$, $\boldsymbol{w} \leftarrow \boldsymbol{w} + \boldsymbol{r}\left(\boldsymbol{r}^{\mathrm{H}}\boldsymbol{r}\right) / \left(\boldsymbol{r}^{\mathrm{H}}\widehat{\boldsymbol{R}}\boldsymbol{r} + 2^{-52}\right)$
5. Output current weight: $\boldsymbol{w}[n] \leftarrow \boldsymbol{w}$
6. Next time index: $n \leftarrow n+1$
7. Round ribbon loop: goto step 2.

5.11. SDP: Determining η

If $R[n]$ and $p[n]$ are changing slowly $\to$ choose η close to 1

If $R[n]$ and $p[n]$ remain const for N_1 time slots, but have completely changed after $N_2 > N_1$:

$\eta^{N_2} = \gamma \ll 1$ and $\eta^{N_1} = 1 - \gamma$

$\to \eta = x^{1/N_1}, \quad$ where $\quad x^{N_2/N_1} + x - 1 = 0, \quad 0 < x < 1$

$N_2/N_1 = 30$ is a reasonable choice in practice $\to \eta = \exp\left(-\frac{2.5}{N_2}\right)$

Radio Communications:

N_2 is the number of samples in which the receiver has moved by λ. For sample rate (bandwidth) B, wavelength λ, and speed v:

$$\eta = \exp\left(-2.5\frac{v}{B\lambda}\right)$$

5.12. Least Mean Square (LMS)

Special Case of SDP for $\eta = 0$:

$$\widehat{\boldsymbol{R}}[n] = \boldsymbol{u}[n]\boldsymbol{u}^{\mathrm{H}}[n], \quad \widehat{\boldsymbol{p}}[n] = \boldsymbol{u}[n]d^{*}[n]$$

$$\mu = \frac{\boldsymbol{r}^{\mathrm{H}}\boldsymbol{r}}{\boldsymbol{r}^{\mathrm{H}}\widehat{\boldsymbol{R}}\boldsymbol{r}} = \frac{1}{\|\boldsymbol{u}[n]\|_2^2}$$

$$\boldsymbol{r} = \widehat{\boldsymbol{p}} - \widehat{\boldsymbol{R}}\boldsymbol{w} = \boldsymbol{u}[n]\underbrace{(d^{*}[n] - \overbrace{\boldsymbol{u}^{\mathrm{H}}[n]\boldsymbol{w}}^{y^{*}[n]})}_{e^{*}[n]} = \boldsymbol{u}[n]e^{*}[n]$$

LEAST MEAN SQUARE (LMS)

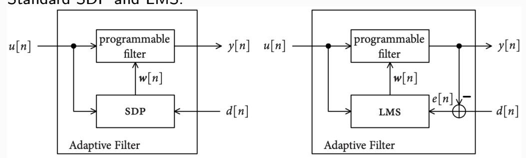
1. Init: $\boldsymbol{w} \leftarrow \mathbf{0}_M$, $n \leftarrow 0$
2. Update weight: If $e[n]$ available: $\boldsymbol{w} \leftarrow \boldsymbol{w} + \boldsymbol{u}[n]\dfrac{e^{*}[n]}{\alpha + \|\boldsymbol{u}[n]\|^2}\beta$
3. Output current weight: $\boldsymbol{w}[n] \leftarrow \boldsymbol{w}$
4. Next time index: $n \leftarrow n+1$
5. Round ribbon loop: goto step 2.

$\alpha > 0$: avoids very large step sizes, if $\boldsymbol{u}[n]$ is small

$\beta > 0$: is for fine-tuning (found experimentally as well as α)

5.13. Block diagrams

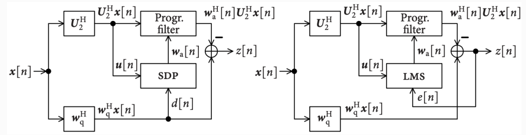
Standard SDP and LMS:



Constrained SDP and LMS:

$$z[n] = \boldsymbol{w}^{\mathrm{H}}[n]\boldsymbol{x}[n]$$

$$\min_{\boldsymbol{w}[n]} \mathrm{E}\left[|z[n]|^2\right], \quad \text{s.t.} \quad \boldsymbol{B}^{\mathrm{H}}\boldsymbol{w}[n] = \boldsymbol{c}$$



$$\boldsymbol{x}[n] = s_i[n]\boldsymbol{b}_i \quad \Longrightarrow \quad z[n] = s_i[n]c_i^{*}$$

$$\boldsymbol{x}[n] \notin \operatorname{im} \boldsymbol{B} \quad \Longrightarrow \quad z[n] = 0$$

6. Least Squares

6.1. Least Squares
Only $\boldsymbol{A}$ is known, no information about $\boldsymbol{\Upsilon}$ $\to$ ignore it all together
Least squares problem:

$$\boldsymbol{X} \approx \boldsymbol{A}\boldsymbol{S}$$
$$\hat{\boldsymbol{S}}_{\mathrm{LS}} = \arg\min_{\boldsymbol{S}} \|\boldsymbol{X} - \boldsymbol{A}\boldsymbol{S}\|_{\mathrm{F}}^2$$

Derivation:

$$\mathcal{E} = \|\boldsymbol{X} - \boldsymbol{A}\boldsymbol{S}\|_{\mathrm{F}}^2 = \operatorname{tr}\left((\boldsymbol{X} - \boldsymbol{A}\boldsymbol{S})^{\mathrm{H}}(\boldsymbol{X} - \boldsymbol{A}\boldsymbol{S})\right)$$

$$\frac{\partial \mathcal{E}}{\partial \boldsymbol{S}^*} = -\boldsymbol{A}^{\mathrm{H}}\boldsymbol{X} + \boldsymbol{A}^{\mathrm{H}}\boldsymbol{A}\boldsymbol{S} \leftarrow \frac{\partial \operatorname{tr}\left(\boldsymbol{S}^{\mathrm{H}}\boldsymbol{B}\right)}{\partial \boldsymbol{S}^*} = \boldsymbol{B}$$

$$\begin{aligned}\hat{\boldsymbol{S}}_{\mathrm{LS}} &= \left(\boldsymbol{A}^{\mathrm{H}}\boldsymbol{A}\right)^{-1}\boldsymbol{A}^{\mathrm{H}}\boldsymbol{X} \text{ (full column rank)}\\ &= \boldsymbol{A}^{+}\boldsymbol{X} = \boldsymbol{A}^{+}\boldsymbol{A}\boldsymbol{S} + \boldsymbol{A}^{+}\boldsymbol{\Upsilon} \text{ (general)}\end{aligned}$$

$$\boldsymbol{A} \text{ has full column rank } \Longrightarrow \hat{\boldsymbol{S}}_{\mathrm{LS}} = \boldsymbol{S} + \boldsymbol{A}^{+}\boldsymbol{\Upsilon}$$
$$\mathrm{E}[\boldsymbol{\Upsilon}] = \boldsymbol{0} \Longrightarrow \text{ unbiased}$$

Estimation noise:

$$\mathrm{E}\left[\left\|\boldsymbol{A}^{+}\boldsymbol{\Upsilon}\right\|_{\mathrm{F}}^2\right] = \operatorname{tr}\left(\boldsymbol{A}^{+}\mathrm{E}\left[\boldsymbol{\Upsilon}\boldsymbol{\Upsilon}^{\mathrm{H}}\right]\boldsymbol{A}^{+\mathrm{H}}\right)$$

For white noise $\mathrm{E}\left[\boldsymbol{\Upsilon}\boldsymbol{\Upsilon}^{\mathrm{H}}\right] = \mathbf{I}$ and full column rank $\boldsymbol{A}$:

$$\mathrm{E}\left[\left\|\boldsymbol{A}^{+}\boldsymbol{\Upsilon}\right\|_{\mathrm{F}}^2\right] = \operatorname{tr}\left(\left(\boldsymbol{A}^{\mathrm{H}}\boldsymbol{A}\right)^{-1}\right) = \sum_i \frac{1}{\lambda_i},$$

where λ_i are the eigenvalues of $\boldsymbol{A}^H\boldsymbol{A}$ $\to$ minimal if all are the same

For white observation noise the smallest possible variance is achieved iff:

$$\boldsymbol{A}^{\mathrm{H}}\boldsymbol{A} = c\mathbf{I}, \quad \text{for any } c > 0$$

among all matrices $\boldsymbol{A} \in \mathbb{C}^{M\times d}$ with $\|\boldsymbol{A}\|_{\mathrm{F}}^2 = c$.

- $\boldsymbol{A}^{\mathrm{H}}\boldsymbol{A}$ has $\lambda_i = c/d$
- $\boldsymbol{A}$ must have orthogonal columns with the same euclidean norm
- use orthogonal pilot sequences
- purely deterministic approach, no need to know statistical properties

6.2. Pilot Sequence
Setup:

$$\boldsymbol{u} = \boldsymbol{H}\boldsymbol{p} + \boldsymbol{v} \iff \boldsymbol{u} = \boldsymbol{A}\boldsymbol{h} + \boldsymbol{v}$$

$$\boldsymbol{A} = \begin{bmatrix} P_0 & P_1 & \cdots & P_K \\ P_1 & P_2 & \cdots & P_{K+1} \\ \vdots & \vdots & \vdots & \vdots \\ P_{q-K-1} & P_{q-K} & \cdots & P_{q-1} \end{bmatrix}$$

$$\boldsymbol{h} = \begin{bmatrix} h_0 & h_1 & \cdots & h_K \end{bmatrix}^{\mathrm{T}}$$

Necessary for full column rank: $q \geq 2K + 1$

6.3. LS curve fitting

$$\hat{y}(x) = \frac{a}{x} + b + cx + dx^2 + ex^3$$

$$\underbrace{\begin{bmatrix} 1/x_1 & 1 & x_1 & x_1^2 & x_1^3 \\ 1/x_2 & 1 & x_2 & x_2^2 & x_2^3 \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ 1/x_N & 1 & x_N & x_N^2 & x_N^3 \end{bmatrix}}_{\boldsymbol{A}} \underbrace{\begin{bmatrix} a \\ b \\ c \\ d \\ e \end{bmatrix}}_{\boldsymbol{w}} \approx \underbrace{\begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_N \end{bmatrix}}_{\boldsymbol{y}}$$

$$\boldsymbol{w}_{\mathrm{LS}} = \boldsymbol{A}^{+}\boldsymbol{y}$$

6.4. Numerical integration with LS
Within every 3x3 window:

$$\hat{f}(x, y) = a + bx^2 + cy^2 + dx^2y^2$$

$$\begin{bmatrix} 1 & h_x^2 & h_y^2 & h_x^2h_y^2 \\ 1 & 0 & h_y^2 & 0 \\ \vdots & \vdots & \vdots & \vdots \\ 1 & h_x^2 & h_y^2 & h_x^2h_y^2 \end{bmatrix} \underbrace{\begin{bmatrix} a \\ b \\ c \\ d \end{bmatrix}}_{w} \approx \begin{bmatrix} f(-h_x, h_y) \\ f(0, h_y) \\ \vdots \\ f(h_x, -h_y) \end{bmatrix}$$

$$\hat{F} = \frac{4}{9}h_xh_y\left(9a + 3bh_x^2 + \left(3c + dh_x^2\right)h_y^2\right)$$

And without solving $w_{\mathrm{LS}} = \boldsymbol{A}^{+}\boldsymbol{f}$:

$$\begin{aligned}\hat{F} = &\left(f(-h_x, h_y) + f(h_x, h_y) + f(-h_x, -h_y) + f(h_x, -h_y)\right)\\ &+ 4\left(f(0, h_y) + f(-h_x, 0) + f(h_x, 0) + f(0, -h_y)\right)\\ &+ 16f(0,0)\right)\frac{h_xh_y}{9}\end{aligned}$$

3x3 window: quadratic convergence when increasing M
5x5 window: cubic convergence when increasing M

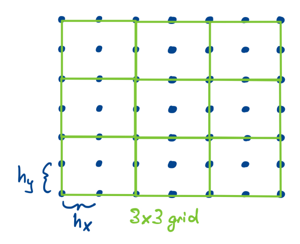
Procedure:

$$F = \int_{y=y_{\min}}^{y_{\max}} \int_{x=x_{\min}}^{x_{\max}} f(x, y)\mathrm{d}x\,\mathrm{d}y$$

$$h_x = \frac{x_{\max} - x_{\min}}{M - 1}, \quad h_y = \frac{y_{\max} - y_{\min}}{M - 1}$$

3x3 window: $M \in \{3, 5, 7, \ldots\}$, 5x5 window: $M \in \{5, 9, 13, \ldots\}$

Divide the integration region into $M{\times}M$ parts. Move the local approximation by $3 - 1$, or $5 - 1$ grid points and sum up all.



6.5. Least squares as a projection
Setup:

$$\boldsymbol{A}\boldsymbol{x} = \boldsymbol{b}, \quad \text{where} \quad \boldsymbol{b} \notin \operatorname{im}\boldsymbol{A}$$
$$\boldsymbol{A}\boldsymbol{x} = \boldsymbol{b} + \Delta\boldsymbol{b}, \quad \text{where} \quad \boldsymbol{b} + \Delta\boldsymbol{b} \in \operatorname{im}\boldsymbol{A}$$
$$\Delta\boldsymbol{b}_{\mathrm{opt}} = \arg\min_{\Delta\boldsymbol{b}} \|\Delta\boldsymbol{b}\|_2^2, \quad \text{s.t.} \quad \boldsymbol{b} + \Delta\boldsymbol{b} \in \operatorname{im}\boldsymbol{A}$$

Derivation:

$$\Delta\boldsymbol{b}_{\mathrm{opt}} = \arg\min_{\Delta\boldsymbol{b}} \|\Delta\boldsymbol{b}\|_2^2, \quad \text{s.t.} \quad \boldsymbol{P}(\boldsymbol{b} + \boldsymbol{\Delta}\boldsymbol{b}) = \boldsymbol{b} + \boldsymbol{\Delta}\boldsymbol{b}$$

$$\Delta\boldsymbol{b}_{\mathrm{opt}} = \arg\min_{\Delta\boldsymbol{b}} \|\Delta\boldsymbol{b}\|_2^2, \quad \text{s.t.} \quad (\mathbf{I} - \boldsymbol{P})(\boldsymbol{b} + \Delta\boldsymbol{b}) = \boldsymbol{0}$$

Lagrangian optimization yields:

$$\Delta\boldsymbol{b} = -(\mathbf{I} - \boldsymbol{P})\boldsymbol{b}$$

The least squares solution $\boldsymbol{x} = \boldsymbol{A}^{+}\boldsymbol{b}$ is the exact solution of $\boldsymbol{A}\boldsymbol{x} = \boldsymbol{P}\boldsymbol{b}$. I.e. the least squares estimation projects the measurements on $\operatorname{im}\boldsymbol{A}$
$$\boldsymbol{P} = \boldsymbol{A}\boldsymbol{A}^{+}$$

6.6. Total Least Squares
Setup:

$$\min \left\|\begin{bmatrix} \Delta\boldsymbol{A} & \Delta\boldsymbol{b} \end{bmatrix}\right\|_F^2 \quad \text{s.t.} \quad (\boldsymbol{A} + \Delta\boldsymbol{A})\boldsymbol{x} = \boldsymbol{b} + \Delta\boldsymbol{b}$$

Rewrite:

$$\begin{bmatrix} \boldsymbol{A} & \boldsymbol{b} \end{bmatrix}\begin{bmatrix} x \\ -1 \end{bmatrix} = 0, \quad \boldsymbol{b} \notin \operatorname{im}\boldsymbol{A}$$

$\boldsymbol{b} \notin \operatorname{im}\boldsymbol{A}$, thus: $\operatorname{rank}[\boldsymbol{A}\boldsymbol{b}] = N + 1$ and $\operatorname{null}[\boldsymbol{A}\boldsymbol{b}] = \{0\}$

$$\begin{bmatrix} \boldsymbol{A} & \boldsymbol{b} \end{bmatrix} = \boldsymbol{U}\boldsymbol{\Sigma}\boldsymbol{V}^{\mathrm{H}} = \sum_{k=1}^{N+1} s_k\boldsymbol{u}_k\boldsymbol{v}_k^{\mathrm{H}}$$

Allow for solutions by increasing the dimensionality of the nullspace to 1:

$$\begin{bmatrix} \boldsymbol{A} + \Delta\boldsymbol{A} & \boldsymbol{b} + \Delta\boldsymbol{b} \end{bmatrix} = \sum_{k=1}^{N} s_k\boldsymbol{u}_k\boldsymbol{v}_k^{\mathrm{H}}$$

$$\begin{bmatrix} \boldsymbol{A} + \Delta\boldsymbol{A} & \boldsymbol{b} + \Delta\boldsymbol{b} \end{bmatrix}\boldsymbol{v}_{N+1}\alpha = \boldsymbol{0}$$

The solution is optimal as we used the best approximation by Eckart-Young

$$\boldsymbol{x} = \frac{-\begin{bmatrix} \mathbf{I}_N & \mathbf{0}_N \end{bmatrix}\boldsymbol{v}_{N+1}}{\begin{bmatrix} \mathbf{0}_N^{\mathrm{T}} & 1 \end{bmatrix}\boldsymbol{v}_{N+1}}$$

7. BLUE

7.1. Best Linear Unbiased Estimator (BLUE)
$\boldsymbol{A}$ and $\mathrm{E}\left[\boldsymbol{\Upsilon}\boldsymbol{\Upsilon}^{\mathrm{H}}\right]$ are known
Setup:

$$\boldsymbol{X} = \boldsymbol{A}\boldsymbol{S} + \boldsymbol{\Upsilon}$$

$$\boldsymbol{W}_{\mathrm{BLUE}} = \arg\min_{\boldsymbol{W}} \mathrm{E}\left[\left\|\boldsymbol{W}^{\mathrm{H}}\boldsymbol{\Upsilon}\right\|_{\mathrm{F}}^2\right] \quad \text{s.t.} \quad \boldsymbol{W}^{\mathrm{H}}\boldsymbol{A} = \mathbf{I}$$

$\to$ minimize estimation variance while being unbiased
Derivation:

$$\mathrm{E}\left[\left\|\boldsymbol{W}^{\mathrm{H}}\boldsymbol{\Upsilon}\right\|_{\mathrm{F}}^2\right] = \operatorname{tr}\left(W^{\mathrm{H}}\mathrm{E}\left[\boldsymbol{\Upsilon}\boldsymbol{\Upsilon}^{\mathrm{H}}\right]\boldsymbol{W}\right)$$

$$\boldsymbol{W}^{\mathrm{H}}\boldsymbol{A} = \mathbf{I} \iff \forall i : \boldsymbol{w}_i^{\mathrm{H}}\boldsymbol{A} = \mathbf{e}_i^{\mathrm{T}}$$

Lagrangian optimization yields:

optimal weights:
$$\boldsymbol{W}_{\mathrm{BLUE}} = \left(\mathrm{E}\left[\boldsymbol{\Upsilon}\boldsymbol{\Upsilon}^{\mathrm{H}}\right]\right)^{-1}\boldsymbol{A}\left(\boldsymbol{A}^{\mathrm{H}}\left(\mathrm{E}\left[\boldsymbol{\Upsilon}\boldsymbol{\Upsilon}^{\mathrm{H}}\right]\right)^{-1}\boldsymbol{A}\right)^{-1}$$
minimal noise variance:
$$\mathrm{E}\left[\left\|\boldsymbol{W}_{\mathrm{BLUE}}^{\mathrm{H}}\boldsymbol{\Upsilon}\right\|_{\mathrm{F}}^2\right] = \operatorname{tr}\left(\left(\boldsymbol{A}^{\mathrm{H}}\left(\mathrm{E}\left[\boldsymbol{\Upsilon}\boldsymbol{\Upsilon}^{\mathrm{H}}\right]\right)^{-1}\boldsymbol{A}\right)^{-1}\right)$$

Special case: $\mathrm{E}\left[\boldsymbol{\Upsilon}\boldsymbol{\Upsilon}^{\mathrm{H}}\right] = \mathbf{I}\sigma_v^2$ white observation noise:

$$\boldsymbol{W}_{\mathrm{BLUE}}^{\mathrm{H}} = \boldsymbol{A}^{+} = \boldsymbol{W}_{\mathrm{LS}}^{\mathrm{H}}$$

- noise must be colored in order for BLUE to improve the result
- $\boldsymbol{S} = 0$ must be transmitted to estimate $\mathrm{E}\left[\boldsymbol{\Upsilon}\boldsymbol{\Upsilon}^{\mathrm{H}}\right]$ $\to$ extra payload
- if noise is WSS, $\mathrm{E}\left[\boldsymbol{\Upsilon}\boldsymbol{\Upsilon}^{\mathrm{H}}\right]$ can be estimated once
- if noise is not WSS: tradeoff between tracking performance and estimation error

7.2. BLUE for uncorrelated signal and noise
Special case: noise and signal are uncorrelated:

For $\mathrm{E}\left[\boldsymbol{S}\boldsymbol{\Upsilon}^{\mathrm{H}}\right] = 0$:
$$\boldsymbol{W}_{\mathrm{BLUE}} = \left(\mathrm{E}\left[\boldsymbol{X}\boldsymbol{X}^{\mathrm{H}}\right]\right)^{-1}\boldsymbol{A}\left(\boldsymbol{A}^{\mathrm{H}}\left(\mathrm{E}\left[\boldsymbol{X}\boldsymbol{X}^{\mathrm{H}}\right]\right)^{-1}\boldsymbol{A}\right)^{-1}$$

If the statistics are unknown, they can be estimated using the observation matrix $\boldsymbol{X}$:

$$\widehat{\boldsymbol{W}}_{\mathrm{BLUE}} = \left(\boldsymbol{X}\boldsymbol{X}^{\mathrm{H}}\right)^{-1}\boldsymbol{A}\left(\boldsymbol{A}^{\mathrm{H}}\left(\boldsymbol{X}\boldsymbol{X}^{\mathrm{H}}\right)^{-1}\boldsymbol{A}\right)^{-1}$$

- no need to switch of signal for estimating the noise variance
- if noise is WSS: estimation is easy (once)
- if noise is not WSS: tradeoff between estimation error and tracking performance
- XX^H must be invertible $\to$ needs enough samples
- loss of generality
- latency: samples $\boldsymbol{X}$ have to be gathered before estimation can start (LS can start immideately)

7.3. BLUE for multi-user detection
General Setup:
Q users with signals $\tilde{\boldsymbol{S}}_i$ and full column rank channels $\tilde{H}_i$, additive white noise, signals S_i and S_j are uncorrelated for $i \neq j$

$$\boldsymbol{X} = \sum_{i=1}^{Q} \tilde{\boldsymbol{H}}_i\tilde{\boldsymbol{S}}_i + \boldsymbol{\Theta}, \quad \mathrm{E}\left[\boldsymbol{\Theta}\boldsymbol{\Theta}^{\mathrm{H}}\right] = \mathbf{I}, \quad \mathrm{E}\left[\tilde{\boldsymbol{S}}_i\tilde{\boldsymbol{S}}_i^{\mathrm{H}}\right] = \sigma_i^2\mathbf{I}$$

Normalized Setup: $\boldsymbol{H}_i = \tilde{\boldsymbol{H}}_i\sigma_i$, and $\boldsymbol{S}_i = \tilde{\boldsymbol{S}}_i/\sigma_i$

$$\boldsymbol{X} = \sum_{i=1}^{Q} \boldsymbol{H}_i\boldsymbol{S}_i + \boldsymbol{\Theta}, \quad \mathrm{E}\left[\boldsymbol{\Theta}\boldsymbol{\Theta}^{\mathrm{H}}\right] = \mathbf{I}, \quad \mathrm{E}\left[\boldsymbol{S}_i\boldsymbol{S}_i^{\mathrm{H}}\right] = \mathrm{I}$$

For reconstructing the signal of the k-th user:

$$\boldsymbol{X} = \boldsymbol{H}_k\boldsymbol{S}_k + \underbrace{\sum_{i=1, i\neq k}^{Q} \boldsymbol{H}_i\boldsymbol{S}_i + \boldsymbol{\Theta}}_{\boldsymbol{\Upsilon}_k} = \boldsymbol{H}_k\boldsymbol{S}_k + \boldsymbol{\Upsilon}_k$$

Solution for the BLUE:

$$\boldsymbol{W}_k = \left(\mathrm{E}\left[\boldsymbol{X}\boldsymbol{X}^{\mathrm{H}}\right]\right)^{-1}\boldsymbol{H}_k\left(\boldsymbol{H}_k^{\mathrm{H}}\left(\mathrm{E}\left[\boldsymbol{X}\boldsymbol{X}^{\mathrm{H}}\right]\right)^{-1}\boldsymbol{H}_k\right)^{-1}$$

$$\mathrm{E}\left[\boldsymbol{X}\boldsymbol{X}^{\mathrm{H}}\right] = \mathbf{I} + \sum_{i=1}^{Q} \boldsymbol{H}_i\boldsymbol{H}_i^{\mathrm{H}}$$

Procedure for estimating all signals:
1. Find the signal with the lowest estimation noise, by computing all noises and finding the minimum

$$\begin{aligned}\xi_k &= \mathrm{E}\left[\left\|\boldsymbol{W}_k^{\mathrm{H}}\boldsymbol{\Upsilon}_k\right\|_{\mathrm{F}}^2\right]\\ &= \operatorname{tr}\left(\boldsymbol{W}_k^{\mathrm{H}}\left(\mathbf{I} + \sum_{i=1, i\neq k}^{Q} \boldsymbol{H}_i\boldsymbol{H}_i^{\mathrm{H}}\right)\boldsymbol{W}_k\right)\end{aligned}$$

$$k_* = \arg\min_{k\in\{1,2,\ldots Q\}} \xi_k$$

2. compute the estimated signal and subtract it from the observation

$$\hat{\boldsymbol{S}}_{k_*} = \boldsymbol{W}_{k_*}^{\mathrm{H}}\boldsymbol{X}$$
$$\boldsymbol{X} \leftarrow \boldsymbol{X} - \boldsymbol{H}_{k_*}\boldsymbol{S}_{k_*}$$

3. repeat the procedure with the remaining measurement
- BLUE usually does not yield the minimum MSE (also unbiased)
- optimum filter wrt. MSE is given by: $(\mathrm{E}[\|\boldsymbol{W}^H\boldsymbol{X} - \boldsymbol{S}\|_{\mathrm{F}}^2])$

$$\boldsymbol{W}_{\mathrm{opt}} = \left(\mathrm{E}\left[\boldsymbol{X}\boldsymbol{X}^{\mathrm{H}}\right]\right)^{-1}\mathrm{E}\left[\boldsymbol{X}\boldsymbol{S}^{\mathrm{H}}\right]$$

8. MUSIC

8.1. MUSIC: Multiple Signal Classification

Nothing is known except for algebraic structure

Setup and restrictions:

$$\boldsymbol{X} = \boldsymbol{A}\boldsymbol{S} + \boldsymbol{\Upsilon}$$

$$\boldsymbol{A} = \begin{bmatrix} \boldsymbol{a}(\theta_1) & \boldsymbol{a}(\theta_2) & \cdots & \boldsymbol{a}(\theta_d) \end{bmatrix} \in \mathbb{C}^{M\times d}$$

$$\boldsymbol{a}(\theta) = \begin{bmatrix} a_1(\theta) \\ a_2(\theta) \\ \vdots \\ a_M(\theta) \end{bmatrix}$$

$$\operatorname{rank}\left(\boldsymbol{S} \in \mathbb{C}^{d\times N}\right) = d, \quad \operatorname{rank}(\boldsymbol{A}) = d, d < M$$

$\theta_1, \theta_2, \dots, \theta_M$ pairwise different

$$\Downarrow$$

$\begin{pmatrix} \boldsymbol{a}(\theta_1) & \boldsymbol{a}(\theta_2) & \cdots & \boldsymbol{a}(\theta_M) \end{pmatrix}$ are linearly independent

Goal: find d, θ_i, and $\boldsymbol{S}$

Derivation without noise:

$$\operatorname{im}(\boldsymbol{A}) = \operatorname{im}\left(\boldsymbol{A}\boldsymbol{S}\boldsymbol{S}^{\mathrm{H}}\boldsymbol{A}^{\mathrm{H}}\right)$$

$$\boldsymbol{X}\boldsymbol{X}^{H} = \boldsymbol{A}\boldsymbol{S}\boldsymbol{S}^{\mathrm{H}}\boldsymbol{A}^{\mathrm{H}} = [\underbrace{\boldsymbol{u}_1 \cdots \boldsymbol{u}_d}_{\boldsymbol{U}_1} \underbrace{\boldsymbol{u}_{d+1} \cdots \boldsymbol{u}_M}_{\boldsymbol{U}_2}]\boldsymbol{\Lambda}\boldsymbol{U}^{\mathrm{H}}$$

$$\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_d > 0$$

$$\operatorname{im}(\boldsymbol{A}) = \operatorname{im}(\boldsymbol{U}_1) \to y \in \operatorname{im}(\boldsymbol{A}) \iff \quad \boldsymbol{U}_2^{\mathrm{H}}\boldsymbol{y} = 0$$

$$\forall i \in \{1, 2, \dots, d\}: \quad \boldsymbol{U}_2^{\mathrm{H}}\boldsymbol{a}(\theta_i) = \boldsymbol{0}$$

In summary:

$$\boxed{\left\|\boldsymbol{U}_2^{\mathrm{H}}\boldsymbol{a}(\theta)\right\|_2^2 \begin{cases} = 0 & \text{for } \theta \in \{\theta_1, \theta_2, \dots, \theta_d\} \\ > 0 & \text{else.} \end{cases}}$$

Modification with noise:

$$\mathrm{E}\left[\boldsymbol{\Upsilon}\boldsymbol{\Upsilon}^{\mathrm{H}}\right] = \sigma_{\Upsilon}^2 \mathbf{I}, \quad \text{and} \quad \mathrm{E}\left[\boldsymbol{S}\boldsymbol{\Upsilon}^{\mathrm{H}}\right] = \mathbf{O}$$

$$\mathrm{E}\left[\boldsymbol{X}\boldsymbol{X}^{\mathrm{H}}\right]$$

$$= [\boldsymbol{U}_1\boldsymbol{U}_2]\operatorname{diag}(\lambda_1+\sigma_{\Upsilon}^2, \dots, \lambda_d+\sigma_{\Upsilon}^2, \sigma_{\Upsilon}^2, \dots, \sigma_{\Upsilon}^2)[\boldsymbol{U}_1\boldsymbol{U}_2]^{H}$$

Approximation:

$$\boldsymbol{X}\boldsymbol{X}^{H} = [\hat{\boldsymbol{U}}_1\hat{\boldsymbol{U}}_2]\operatorname{diag}(\hat{\lambda}_1, \dots, \hat{\lambda}_M)[\hat{\boldsymbol{U}}_1\hat{\boldsymbol{U}}_2]^{H}$$

For determining d, find the last significant drop in $\hat{\lambda}_i$.

MUSIC spectrum: has maxima close to θ_i

$$\boxed{F(\theta) = \frac{\|\boldsymbol{a}(\theta)\|_2^2}{\left\|\widehat{\boldsymbol{U}}_2^{\mathrm{H}}\boldsymbol{a}(\theta)\right\|_2^2}}$$

Procedure:

1. Compute eigenvalue decomposition of XX^H
2. Scan eigenvalues for jumps and determine $\hat{d}$
3. Find locations $\hat{\theta}_i$ at $\hat{d}$ strongest peaks of the MUSIC spectrum $F(\theta)$
4. Form the matrix $\hat{\boldsymbol{A}} = \begin{bmatrix} \boldsymbol{a}(\hat{\theta}_1) & \boldsymbol{a}(\hat{\theta}_2) & \cdots & \boldsymbol{a}(\hat{\theta}_{\hat{d}}) \end{bmatrix}$
5. Solve for $\hat{\boldsymbol{S}}$ with pseudo-inverse $\hat{\boldsymbol{S}} = \hat{\boldsymbol{A}}^{+}\boldsymbol{X}$

Signal-to-noise ratio:

$$\mathrm{SNR} = \frac{\mathrm{E}\left[\|\boldsymbol{A}\boldsymbol{S}\|_{\mathrm{F}}^2\right]}{\mathrm{E}\left[\|\boldsymbol{\Upsilon}\|_{\mathrm{F}}^2\right]}$$

Maximum number of signals:
At most $d = M - 1$ complex components can be resolved $\to$ $\boldsymbol{U}_2$ does not exist for larger d. Otherwise, M has to be increased by either lowering N or increasing the number if samples.

Sharpness of the peaks:
The peaks get sharper when N is increased $\to$ $\hat{\boldsymbol{U}}_2$ more accurate. The higher N, the closer targets can be resolved.

MUSIC with colored noise:

$$\begin{aligned} \boldsymbol{X}' &= \left(\mathrm{E}\left[\boldsymbol{\Upsilon}\boldsymbol{\Upsilon}^{\mathrm{H}}\right]\right)^{-1/2}\boldsymbol{X} \\ &= \underbrace{\left(\mathrm{E}\left[\boldsymbol{\Upsilon}\boldsymbol{\Upsilon}^{\mathrm{H}}\right]\right)^{-1/2}\boldsymbol{A}}_{\boldsymbol{A}'}\boldsymbol{S} + \underbrace{\left(\mathrm{E}\left[\boldsymbol{\Upsilon}\boldsymbol{\Upsilon}^{\mathrm{H}}\right]\right)^{-1/2}\boldsymbol{\Upsilon}}_{\boldsymbol{\Upsilon}'} \\ &= \boldsymbol{A}'\boldsymbol{S} + \boldsymbol{\Upsilon}'. \end{aligned}$$

8.2. MUSIC for multiple frequency estimation

Setup:

$$\boldsymbol{a}(\omega_i) = \begin{bmatrix} 1 \\ \mathrm{e}^{\mathrm{j}\omega_i T} \\ \mathrm{e}^{\mathrm{j}2\omega_i T} \\ \vdots \\ \mathrm{e}^{\mathrm{j}(M-1)\omega_i T} \end{bmatrix}$$

$$x[n] = \sum_{i=1}^{d} c_i \mathrm{e}^{\mathrm{j}\omega_i T n}$$

$$\boldsymbol{x}[n] = \begin{bmatrix} x[n] \\ x[n+1] \\ x[n+2] \\ \vdots \\ x[n+M-1] \end{bmatrix} = \sum_{i=1}^{d} \begin{bmatrix} c_i \mathrm{e}^{\mathrm{j}\omega_i T n} \\ c_i \mathrm{e}^{\mathrm{j}\omega_i T(n+1)} \\ c_i \mathrm{e}^{\mathrm{j}\omega_i T(n+2)} \\ \vdots \\ c_i \mathrm{e}^{\mathrm{j}\omega_i T(n+M-1)} \end{bmatrix}$$

$$= \underbrace{\begin{bmatrix} \boldsymbol{a}(\omega_1) & \cdots & \boldsymbol{a}(\omega_d) \end{bmatrix}}_{\boldsymbol{A}} \underbrace{\begin{bmatrix} c_1 \mathrm{e}^{\mathrm{j}\omega_1 T n} \\ c_2 \mathrm{e}^{\mathrm{j}\omega_2 T n} \\ \vdots \\ c_d \mathrm{e}^{\mathrm{j}\omega_d T n} \end{bmatrix}}_{\boldsymbol{s}[n]} = \boldsymbol{A}\boldsymbol{s}[n]$$

$$\boldsymbol{X} = \begin{bmatrix} \boldsymbol{x}[n] & \boldsymbol{x}[n+1] & \cdots & \boldsymbol{x}[n+N-1] \end{bmatrix}$$

$\boldsymbol{A}$ is Vandermode $\to$ for pairwise different θ_i, the columns of $\boldsymbol{A}$ are LI

8.3. Sampling of plane waves

$$p(t, \vec{r}) = P(t - \vec{\kappa} \cdot \vec{r}/c)$$

$\vec{\kappa}$: direction unit vector, c: propagation speed
Harmonic planar wave:

$$p(t, \underline{r}) = A\cos\left(\omega_0 t - k_0 \vec{\kappa} \cdot \vec{r} + \varphi\right), k_0 = \omega_0/c = 2\pi f_0/c$$

Modulated harmonic planar wave:

$$p(t, \vec{r}) = A(t - \vec{\kappa} \cdot \vec{r}/c)\cos\left(\omega_0 t - k_0 \vec{\kappa} \cdot \vec{r} + \varphi(t - \vec{\kappa} \cdot \vec{r}/c)\right)$$

If changes occur only slowly in time (approx. constant withing one period):
$A(t - \lambda_0/c) = A(t - 1/f_0) \approx A(t)$
$\varphi(t - \lambda_0/c) = \varphi(t - 1/f_0) \approx \varphi(t)$

$$p(t, \vec{r} + \vec{\kappa}\lambda_0) = p(t, \vec{r})$$

Short notation with complex numbers:

$$s(t, \vec{r}) = A(t - \vec{\kappa} \cdot \vec{r}/c)\mathrm{e}^{-\mathrm{j}k_0 \vec{\kappa} \cdot \vec{r} + \mathrm{j}\varphi(t - \vec{\kappa} \cdot \vec{r}/c)}$$

$$p(t, \vec{r}) = \operatorname{Re}\left\{s(t, \vec{r})\mathrm{e}^{\mathrm{j}\omega_0 t}\right\}$$

After modulating to the baseband and filtering with a LP:

$$\boldsymbol{x}[n] = \begin{bmatrix} s(nT, \vec{r}_0) \\ s(nT, \vec{r}_1) \\ \vdots \\ s(nT, \vec{r}_{M-1}) \end{bmatrix}$$

If sampling points are close together: $\forall i: \quad |\vec{r}_i| \ll cT, \ (cT/30)$

$$\begin{aligned} x[n] &= \underbrace{A(nT)\mathrm{e}^{j\varphi(nT)}\mathrm{e}^{j\psi}}_{s[n]} \overbrace{\begin{bmatrix} \mathrm{e}^{-j(k_0 \vec{\kappa} \cdot \vec{r}_0 + \psi)} \\ \mathrm{e}^{-j(k_0 \vec{\kappa} \cdot \vec{r}_1 + \psi)} \\ \vdots \\ \mathrm{e}^{-j\left(k_0 \vec{\kappa} \cdot \vec{r}_{M-1} + \psi\right)} \end{bmatrix}}^{\boldsymbol{a}(\vec{\kappa})} \\ &= s[n]\boldsymbol{a}(\vec{\kappa}) \end{aligned}$$

The angle ψ is arbitrary as it cancels out $\to$ used for simplifying

8.4. MUSIC for DOA estimation using ULAs

M sensors on a line with constant distance δ along the z-axis:

$$\vec{r}_i = \left(\frac{1}{2}(M-1) - i\right)\delta\vec{e}_z, \quad i \in \{0, 1, \dots, M-1\}$$

$$\vec{\kappa} = -\cos(\phi)\sin(\theta)\mathbf{e}_x - \sin(\phi)\sin(\theta)\mathbf{e}_y - \cos(\theta)\mathbf{e}_z$$

The angle is measured from the side of the ULA. The steering vector becomes:

$$\boldsymbol{a}(\theta) = \begin{bmatrix} 1 \\ \mathrm{e}^{-\mathrm{j}k_0\delta\cos\theta} \\ \mathrm{e}^{-2\mathrm{j}k_0\delta\cos\theta} \\ \vdots \\ \mathrm{e}^{-(M-1)jk_0\delta\cos\theta} \end{bmatrix}$$

with

$$\psi = \frac{1}{2}(M-1)k_0\delta\cos\theta$$

spacial normalized angular frequency:

$$\mu = -k_0\delta\cos\theta = -2\pi\frac{\delta}{\lambda_0}\cos\theta$$

$$\boldsymbol{a}(\mu) = \begin{bmatrix} 1 \\ \mathrm{e}^{\mathrm{j}\mu} \\ \mathrm{e}^{\mathrm{j}2\mu} \\ \vdots \\ \mathrm{e}^{\mathrm{j}(M-1)\mu} \end{bmatrix}$$

aliasing:

$$\boldsymbol{a}(\mu) = \boldsymbol{a}(\mu + 2\pi n), \quad n \in \{\pm 1, \pm 2, \cdots\}$$

$$\text{no directional aliasing} \iff k_0\delta < \pi \iff \delta < \lambda_0/2$$

If $0°$ is chosen to be orthogonal to the ULA and measured from the center to the right:

$$\boldsymbol{a}(\theta) = \begin{bmatrix} 1 \\ \mathrm{e}^{-\mathrm{j}k_0\delta\sin\theta} \\ \mathrm{e}^{-2\mathrm{j}k_0\delta\sin\theta} \\ \vdots \\ \mathrm{e}^{-(M-1)jk_0\delta\sin\theta} \end{bmatrix}$$

8.5. Useful identities

$$1 + z + z^2 + \cdots + z^{M-1} = \frac{z^M - 1}{z - 1}$$

$$z^M - 1 = 0 \Leftrightarrow z = e^{j2\pi n/M}, \quad n \in \{\pm 1, \pm 2, \dots\}$$

$$\boldsymbol{A}^{-1} = \frac{1}{ad - bc}\begin{pmatrix} d & -b \\ -c & a \end{pmatrix}$$