Polynomial Matrices & Decompositions; Applications to Beamforming and Source Separation

Stephan Weiss

UDRC — Loughborough, Surrey, Strathclyde & Cardiff Consortium
Department of Electonic & Electrical Engineering
University of Strathclyde, Glasgow, Scotland, UK

UDRC Summer School, University of Surrey, 20-23 July 2015

With many thanks to:
John G. McWhirter, Ian K. Proudler, and Jamie Corr

This work was supported by the Engineering and Physical Sciences Research Council (EPSRC) Grant number EP/K014307/1 and the MOD University Defence Research Collaboration in Signal Processing.

Presentation Overview

What is a Polynomial Matrix?

- A polynomial matrix is a polynomial with matrix-valued coefficients, e.g.:

$$\boldsymbol{A}(z) = \begin{bmatrix} 1 & -1 \\ -1 & 2 \end{bmatrix} + \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} z^{-1} + \begin{bmatrix} -1 & 2 \\ 1 & -1 \end{bmatrix} z^{-2} \quad (1)$$

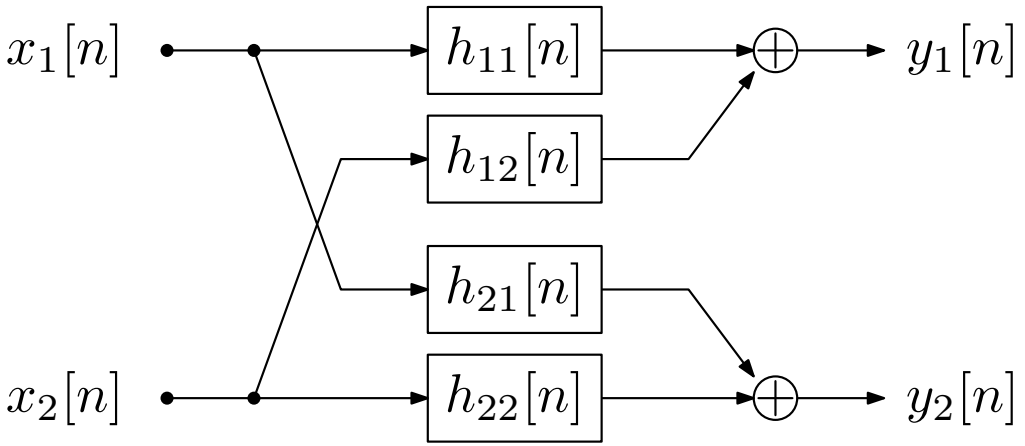
- a polynomial matrix can equivalently be understood a matrix with polynomial entries, i.e.

$$\boldsymbol{A}(z) = \begin{bmatrix} 1 + z^{-1} - z^{-2} & -1 + z^{-1} + 2z^{-2} \\ -1 + z^{-1} + z^{-2} & 2 - z^{-1} - z^{-2} \end{bmatrix} \quad (2)$$

- polynomial matrices could also contain rational polynomials, but the notation would not be easily as interchangeable as (1) and (2).

Where Do Polynomial Matrices Arise?

- A multiple-input multiple-output (MIMO) system could be made up of a number of finite impulse response (FIR) channels:

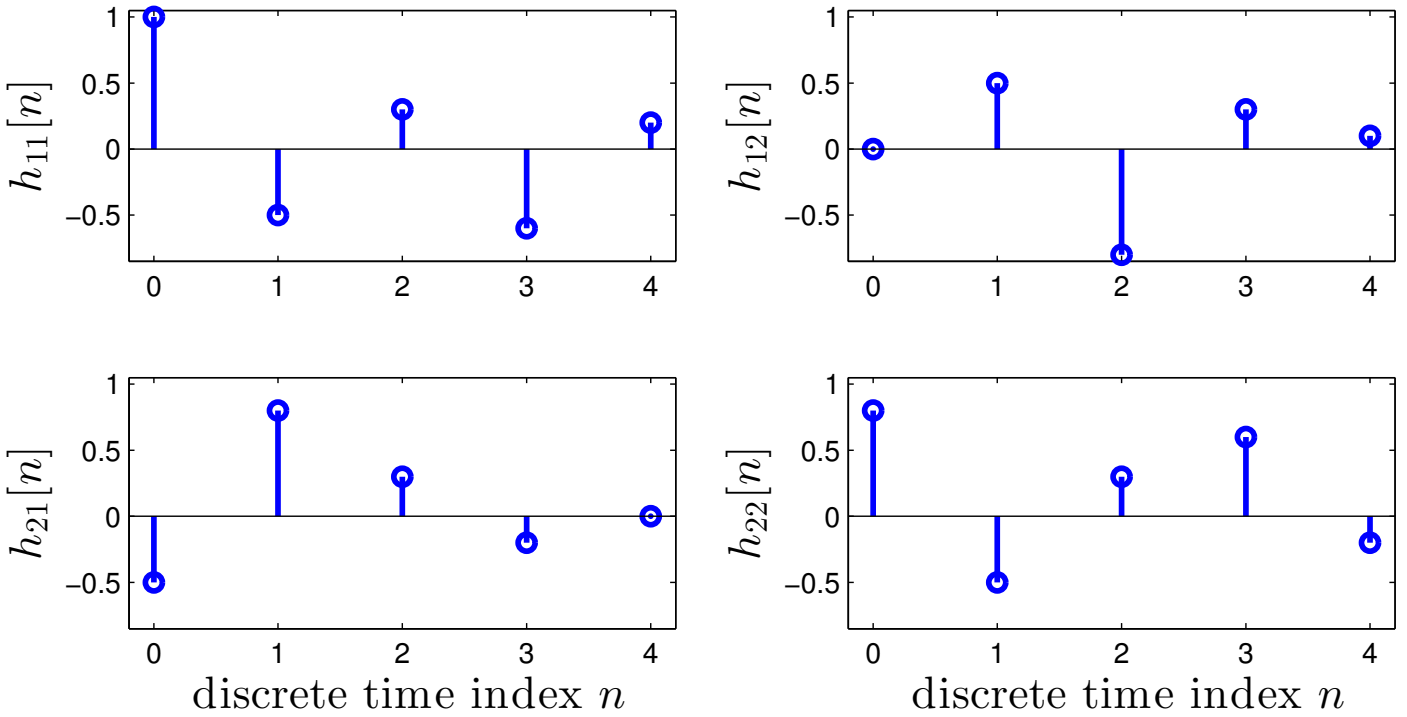


- writing this as a matrix of impulse responses:

$$\mathbf{H}[n] = \begin{bmatrix} h_{11}[n] & h_{12}[n] \\ h_{21}[n] & h_{22}[n] \end{bmatrix} \tag{3}$$

Transfer Function of a MIMO System

- Example for MIMO matrix $\mathbf{H}[n]$ of impulse responses:

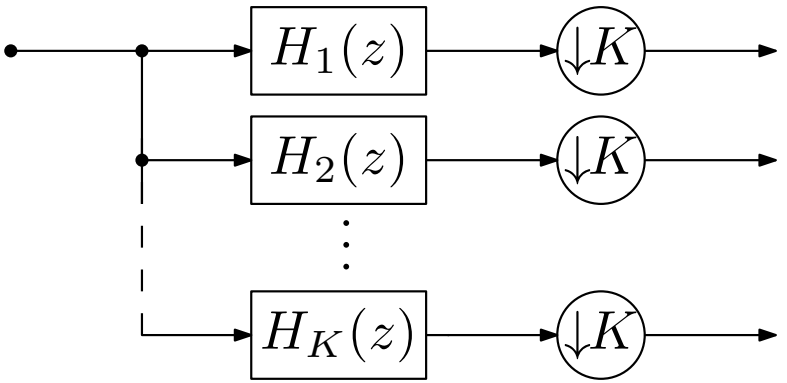


- the transfer function of this MIMO system is a polynomial matrix:

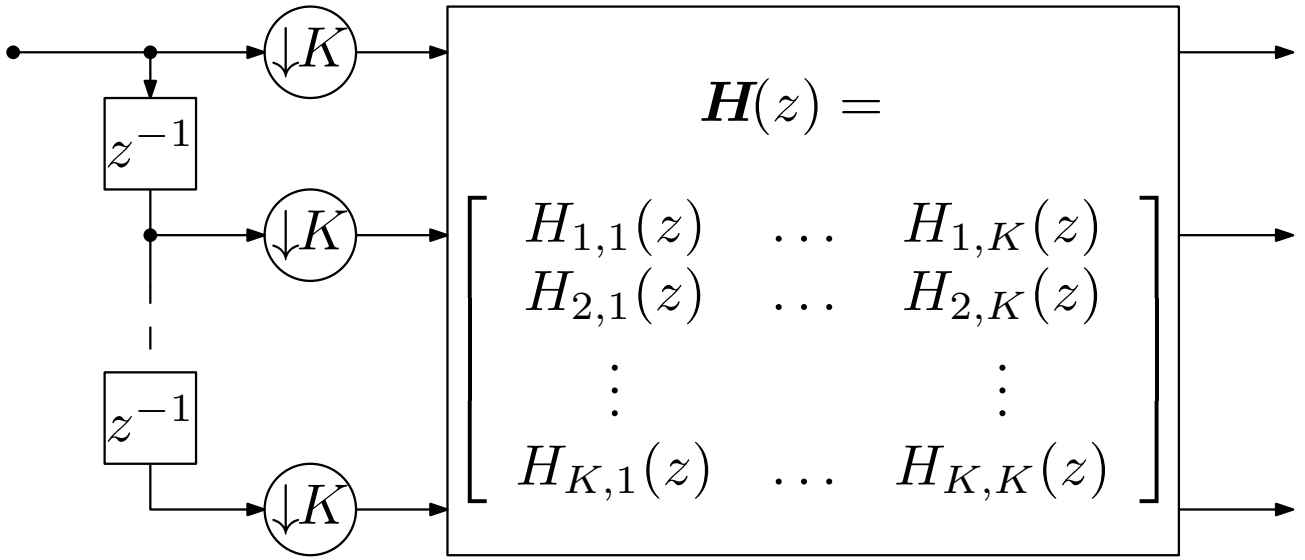
$$\boldsymbol{H}(z) = \sum_{n=-\infty}^{\infty} \mathbf{H}[n] z^{-1} \qquad \text{or} \qquad \boldsymbol{H}(z) \bullet\!\!-\!\!\circ\, \mathbf{H}[n] \tag{4}$$

Analysis Filter Bank

- Critically decimated K-channel analysis filter bank:



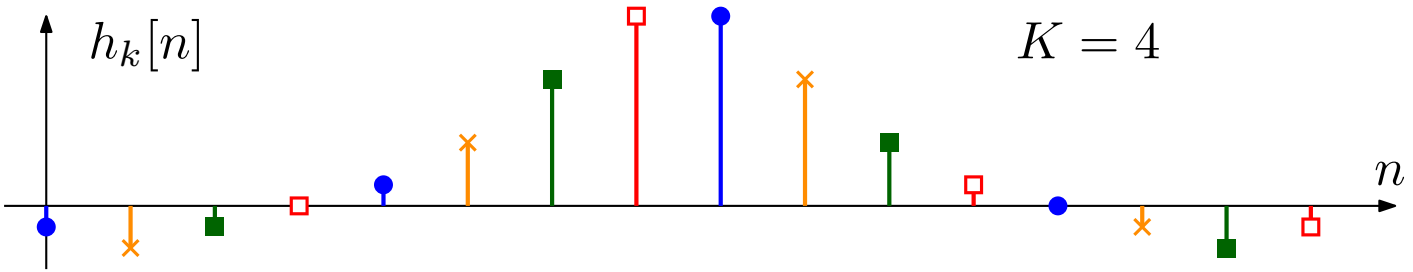
- equivalent polyphase representation:



Polyphase Analysis Matrix

- With the K-fold polyphase decomposition of the analysis filters

$$H_k(z) = \sum_{n=1}^{K} H_{k,n}(z^K) z^{-n+1} \tag{5}$$

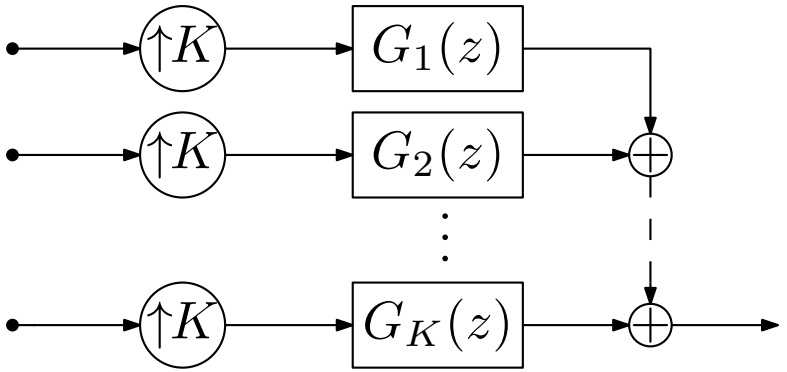


- the polyphase analysis matrix is a polynomial matrix:

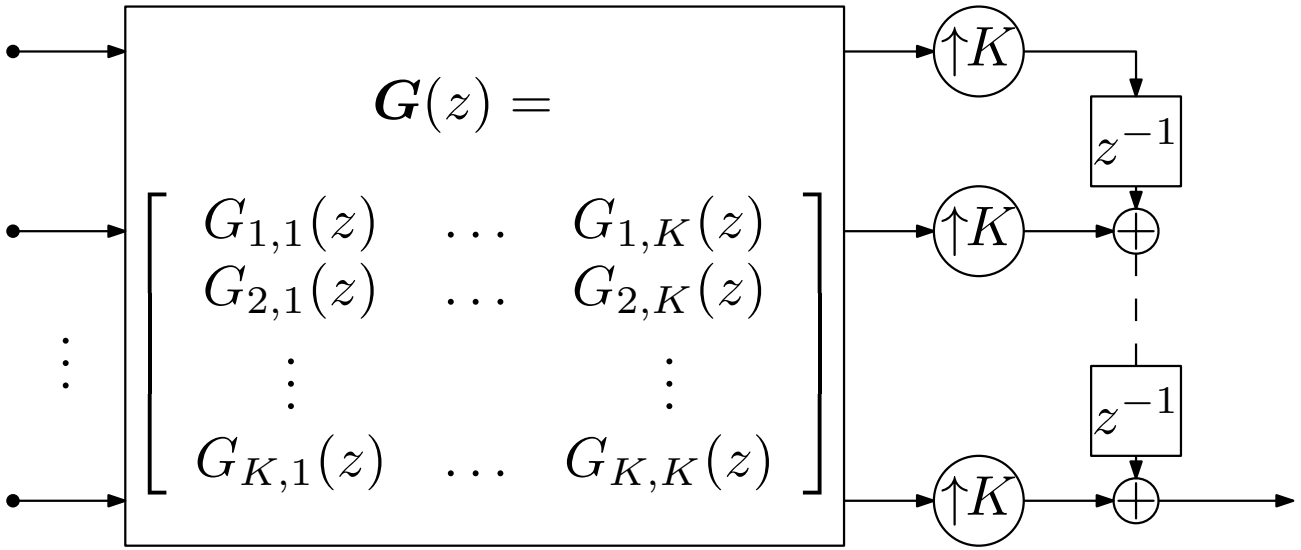
$$\boldsymbol{H}(z) = \begin{bmatrix} H_{1,1}(z) & H_{1,2}(z) & \dots & H_{1,K}(z) \\ H_{2,1}(z) & H_{2,2}(z) & \dots & H_{2,K}(z) \\ \vdots & \vdots & \ddots & \vdots \\ H_{K,1}(z) & H_{K,2}(z) & \dots & H_{K,K}(z) \end{bmatrix} \tag{6}$$

Synthesis Filter Bank

- Critically decimated K-channel synthesis filter bank:



- equivalent polyphase representation:



Polyphase Synthesis Matrix

- Analoguous to analysis filter bank, the synthesis filters $G_k(z)$ can be split into K polyphase components, creating a polyphse synthesis matrix

$$\boldsymbol{G}(z) = \begin{bmatrix} G_{1,1}(z) & G_{1,2}(z) & \dots & G_{1,K}(z) \\ G_{2,1}(z) & G_{2,2}(z) & \dots & G_{2,K}(z) \\ \vdots & \vdots & \ddots & \vdots \\ G_{K,1}(z) & G_{K,2}(z) & \dots & G_{K,K}(z) \end{bmatrix} \tag{7}$$

- operating analysis and synthesis back-to-back, perfect reconstruction is achieved if

$$\boldsymbol{G}(z)\boldsymbol{H}(z) = \mathbf{I}\,; \tag{8}$$

- i.e. for perfect reconstruction, the polyphase analysis matrix must be invertible: $\boldsymbol{G}(z) = \boldsymbol{H}^{-1}(z)$.

Space-Time Covariance Matrix

- Measurements obtained from M sensors are collected in a vector $\mathbf{x}[n] \in \mathbb{C}^M$:

$$\mathbf{x}^{\mathrm{T}}[n] = [x_1[n] \;\; x_2[n] \;\; \ldots \;\; x_M[n]] \;; \tag{9}$$

- with the expectation operator $\mathcal{E}\{\cdot\}$, the spatial correlation is captured by $\mathbf{R} = \mathcal{E}\{\mathbf{x}[n]\mathbf{x}^{\mathrm{H}}[n]\}$;
- for spatial and temporal correlation, we require a space-time covariance matrix

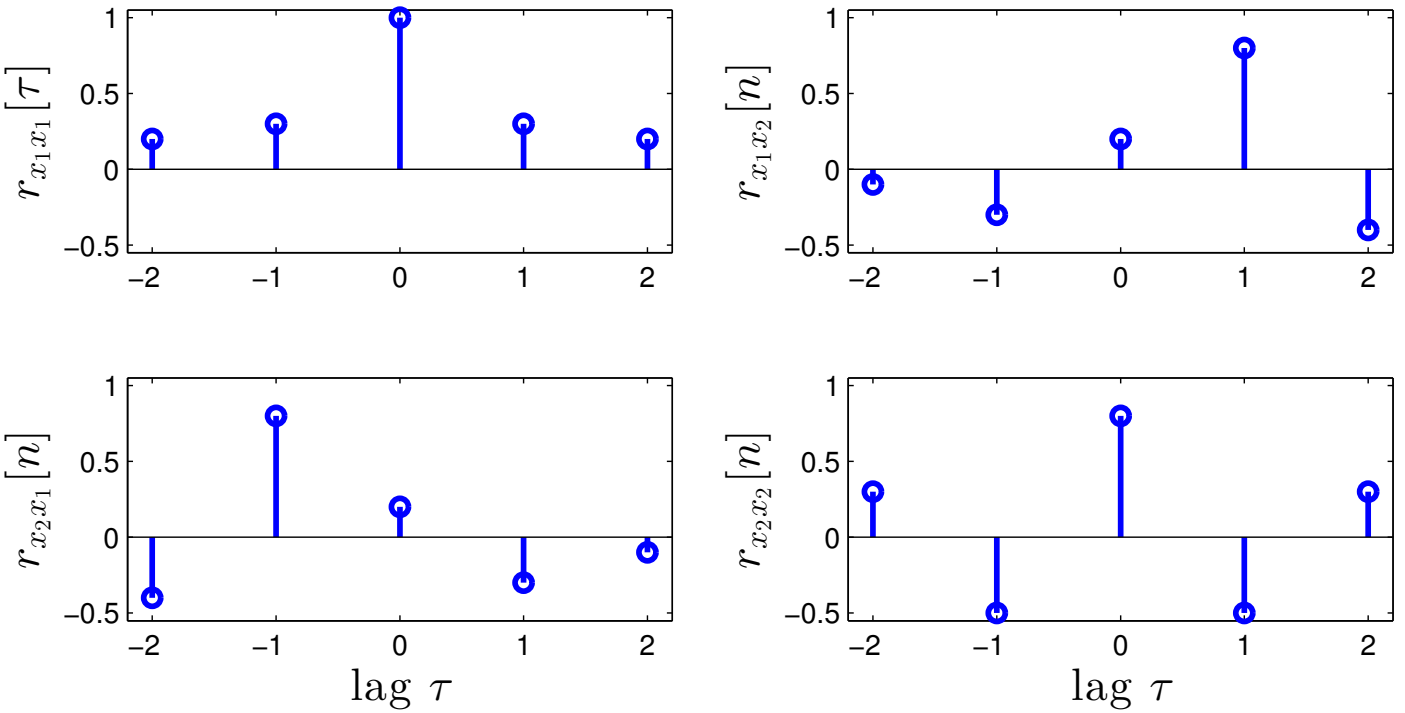
$$\mathbf{R}[\tau] = \mathcal{E}\{\mathbf{x}[n]\mathbf{x}^{\mathrm{H}}[n-\tau]\} \tag{10}$$

- this space-time covariance matrix contains auto- and cross-correlation terms, e.g. for $M = 2$

$$\mathbf{R}[\tau] = \begin{bmatrix} \mathcal{E}\{x_1[n]x_1^*[n-\tau]\} & \mathcal{E}\{x_1[n]x_2^*[n-\tau]\} \\ \mathcal{E}\{x_2[n]x_1^*[n-\tau]\} & \mathcal{E}\{x_2[n]x_2^*[n-\tau]\} \end{bmatrix} \tag{11}$$

Cross-Spectral Density Matrix

- example for a space-time covariance matrix $\mathbf{R}[\tau] \in \mathbb{R}^{2\times 2}$:



- the cross-spectral density (CSD) matrix

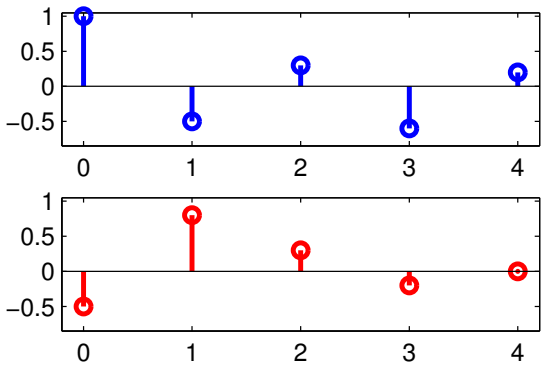
$$\boldsymbol{R}(z) \circ\!\!-\!\!\bullet\, \mathbf{R}[\tau] \tag{12}$$

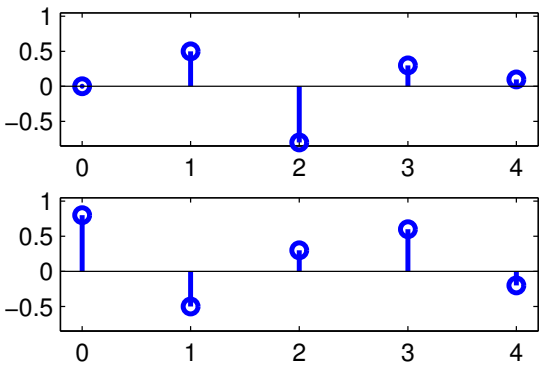
is a polynomial matrix.

Parahermitian Operator

- A parahermitian operation is indicated by $\{\widetilde{\cdot}\}$, and compared to the Hermitian (= complex conjugate transpose) of a matrix additionally performs a time-reversal;
- example:

$$\boldsymbol{A}(z) = \left[\quad \right.$$

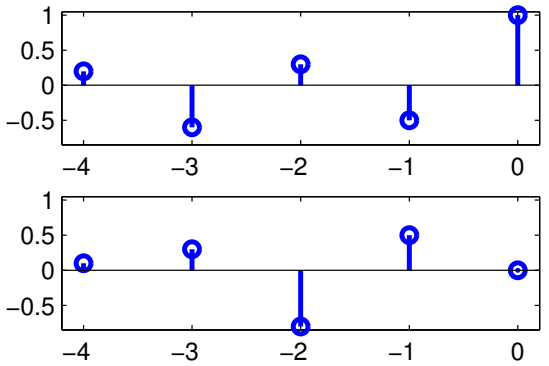


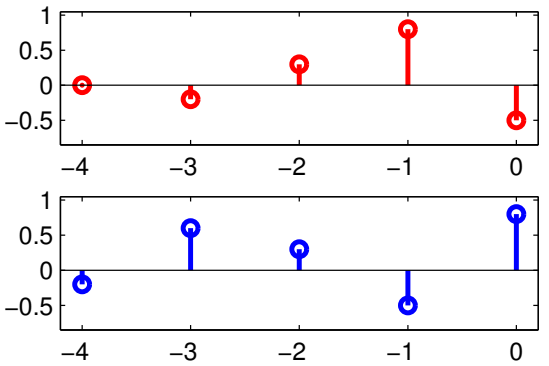


$$\left. \quad \right]$$

- parahermitian $\tilde{\boldsymbol{A}}(z) = \boldsymbol{A}^{\mathrm{H}}(z^{-1})$:

$$\tilde{\boldsymbol{A}}(z) = \left[\quad \right.$$

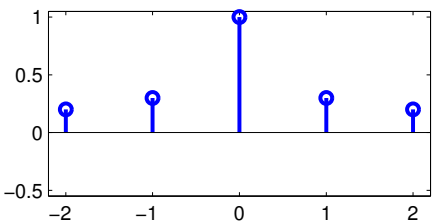
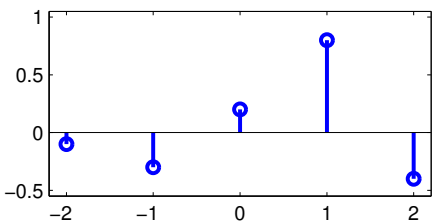
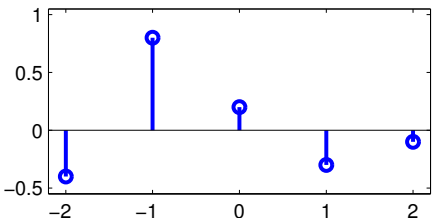
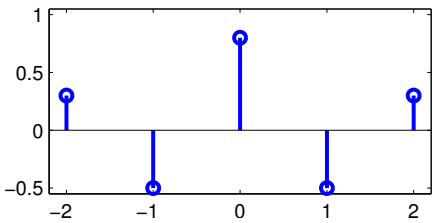




$$\left. \quad \right]$$

Parahermitian Property

- A polynomial matrix $\boldsymbol{A}(z)$ is parahermitian if $\tilde{\boldsymbol{A}}(z) = \boldsymbol{A}(z)$;
- this is an extension of the symmetric (if $\mathbf{A} \in \mathbb{R}$) or or Hermitian (if $\mathbf{A} \in \mathbb{C}$) property to the polynomial case:
  transposition, complex conjugation and time reversal (in any order) do not alter a parahermitian $\boldsymbol{A}(z)$;
- any CSD matrix is parahermitian;
- example:

$$\boldsymbol{R}(z) = \left[\begin{array}{cc} \text{  } & \text{  } \\ \text{  } & \text{  } \end{array} \right] = \tilde{\boldsymbol{R}}(z)$$

Paraunitary Matrices

- Recall that $\mathbf{A} \in \mathbb{C}$ (or $\mathbf{A} \in \mathbb{R}$) is a unitary (or orthonormal) matrix if $\mathbf{A}\mathbf{A}^{\mathrm{H}} = \mathbf{A}^{\mathrm{H}}\mathbf{A} = \mathbf{I}$;
- in the polynomial case, $\mathbf{A}(z)$ is paraunitary if

$$\boldsymbol{A}(z)\tilde{\boldsymbol{A}}(z) = \tilde{\boldsymbol{A}}(z)\boldsymbol{A}(z) = \mathbf{I} \tag{13}$$

- therefore, if $\boldsymbol{A}(z)$ is paraunitary, then the polynomial matrix inverse is simple:

$$\boldsymbol{A}^{-1}(z) = \tilde{\boldsymbol{A}}(z) \tag{14}$$

- example: polyphase analysis or synthesis matrices of perfectly reconstructing (or lossless) filter banks are usually paraunitary.

Attempt of Gaussian Elimination

- System of polynomial equations:

$$\begin{bmatrix} A_{11}(z) & A_{12}(z) \\ A_{21}(z) & A_{22}(z) \end{bmatrix} \cdot \begin{bmatrix} X_1(z) \\ X_2(z) \end{bmatrix} = \begin{bmatrix} B_1(z) \\ B_2(z) \end{bmatrix} \tag{15}$$

- modification of 2nd row:

$$\begin{bmatrix} A_{11}(z) & A_{12}(z) \\ A_{11}(z) & \frac{A_{11}(z)}{A_{21}(z)} A_{22}(z) \end{bmatrix} \cdot \begin{bmatrix} X_1(z) \\ X_2(z) \end{bmatrix} = \begin{bmatrix} B_1(z) \\ \frac{A_{11}(z)}{A_{21}(z)} B_2(z) \end{bmatrix} \tag{16}$$

- upper triangular form by subtracting 1st row from 2nd:

$$\begin{bmatrix} A_{11}(z) & A_{12}(z) \\ 0 & \frac{A_{11}(z)A_{22}(z) - A_{12}(z)A_{21}(z)}{A_{21}(z)} \end{bmatrix} \cdot \begin{bmatrix} X_1(z) \\ X_2(z) \end{bmatrix} = \begin{bmatrix} B_1(z) \\ \bar{B}_2(z) \end{bmatrix} \tag{17}$$

- penalty: we end up with rational polynomials.

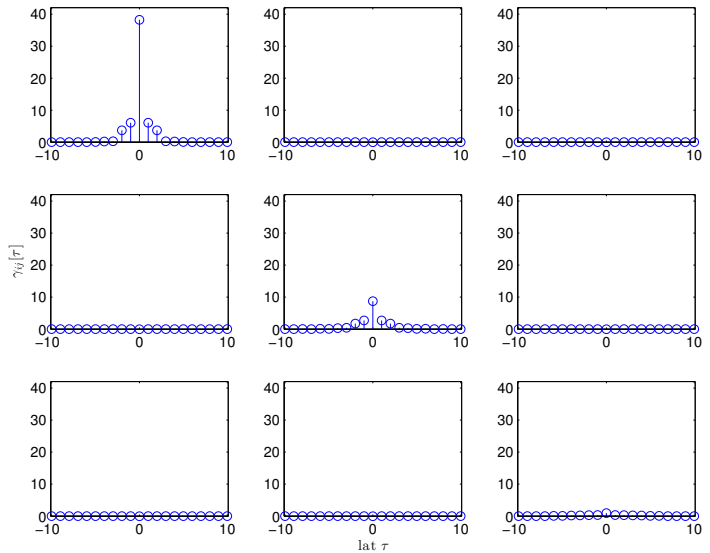
Polynomial Eigenvalue Decomposition

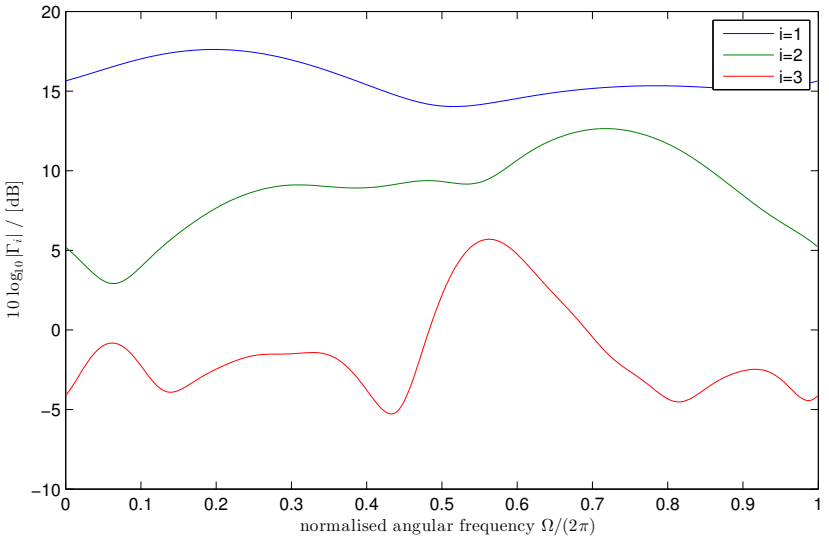
[McWhirter *et al.*, IEEE TSP 2007]

- Polynomial EVD of the CSD matrix

$$\boldsymbol{R}(z) \approx \boldsymbol{Q}(z)\, \boldsymbol{\Lambda}(z)\, \tilde{\boldsymbol{Q}}(z) \tag{18}$$

- with paraunitary $\mathbf{Q}(z)$, s.t. $\mathbf{Q}(z)\tilde{\mathbf{Q}}(z) = \mathbf{I}$;
- diagonalised and spectrally majorised $\boldsymbol{\Lambda}(z)$:





- approximation in (18) can be close with an FIR $\boldsymbol{Q}(z)$ of sufficiently high order [Icart & Comon 2012].

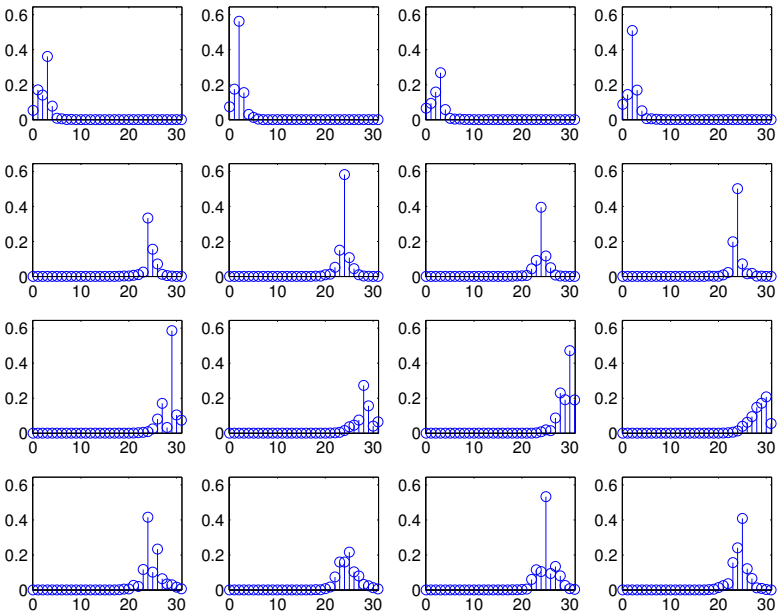
PEVD Ambiguity

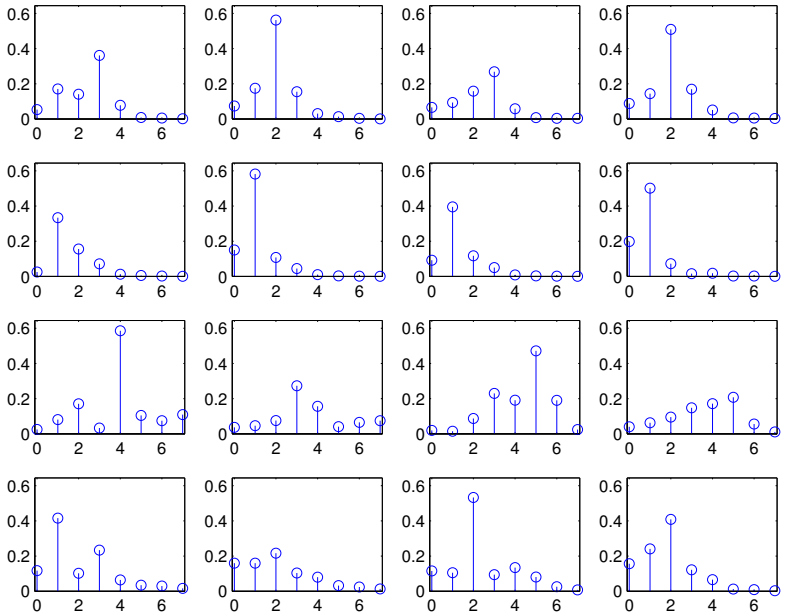
[Corr *et al.*, EUSIPCO 2015]

- We believe diagonalised and spectral majorised $\boldsymbol{\Lambda}(z)$ is unique;
- but there is ambiguity w.r.t. the paraunitary matrix $\boldsymbol{Q}(z)$;
- set $\bar{\boldsymbol{Q}}(z) = \boldsymbol{Q}(z)\boldsymbol{\Gamma}(z)$, with a diagonal allpass $\boldsymbol{\Gamma}(z)$:

$$\begin{aligned}\boldsymbol{R}(z) &= \bar{\boldsymbol{Q}}(z)\boldsymbol{\Lambda}(z)\tilde{\bar{\boldsymbol{Q}}}(z) = \boldsymbol{Q}(z)\boldsymbol{\Gamma}(z)\boldsymbol{\Lambda}(z)\tilde{\boldsymbol{\Gamma}}(z)\tilde{\boldsymbol{Q}}(z) \\ &= \boldsymbol{Q}(z)\boldsymbol{\Lambda}(z)\boldsymbol{\Gamma}(z)\tilde{\boldsymbol{\Gamma}}(z)\tilde{\boldsymbol{Q}}(z) = \boldsymbol{Q}(z)\boldsymbol{\Lambda}(z)\tilde{\boldsymbol{Q}}(z) \qquad (19)\end{aligned}$$

- example for $\tilde{\boldsymbol{Q}}(z)$ — note different orders:





Iterative PEVD Algorithms

- Second order sequential best rotation (SBR2, McWhirter 2007);
- iterative approach based on an elementary paraunitary operation:

$$\begin{aligned} \boldsymbol{S}^{(0)}(z) &= \boldsymbol{R}(z) \\ &\vdots \\ \boldsymbol{S}^{(i+1)}(z) &= \tilde{\boldsymbol{H}}^{(i+1)}(z)\boldsymbol{S}^{(i+1)}(z)\boldsymbol{H}^{(i+1)}(z) \end{aligned}$$

- $\boldsymbol{H}^{(i)}(z)$ is an elementary paraunitary operation, which at the ith step eliminates the largest off-diagonal element in $\boldsymbol{s}^{(i-1)}(z)$;
- stop after L iterations:

$$\hat{\boldsymbol{\Lambda}}(z) = \boldsymbol{S}^{(L)}(z) \qquad , \qquad \boldsymbol{Q}(z) = \prod_{i=1}^{L} \boldsymbol{H}^{(i)}(z)$$

- sequential matrix diagonalisation (SMD) and
- multiple-shift SMD (MS-SMD) will follow the same scheme ...

Elementary Paraunitary Operation

- An elementary paraunitary matrix [Vaidyanathan] is defined as

$$\boldsymbol{H}^{(i)}(z) = \mathbf{I} - \mathbf{v}^{(i)}\mathbf{v}^{(i),\mathrm{H}} + z^{-1}\mathbf{v}^{(i)}\mathbf{v}^{(i),\mathrm{H}} \qquad , \quad \|\mathbf{v}^{(i)}\|_2 = 1$$

- we utilise a different definition:

$$\boldsymbol{H}^{(i)}(z) = \boldsymbol{D}^{(i)}(z)\mathbf{Q}^{(i)}$$

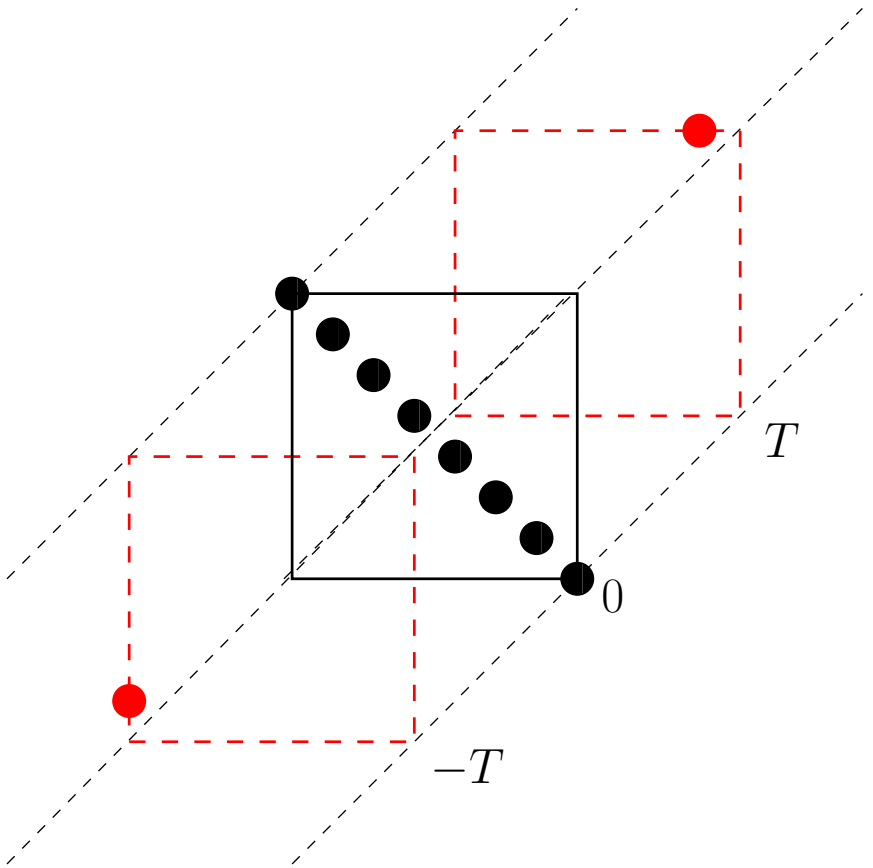
- $\boldsymbol{D}^{(i)}(z)$ is a delay matrix:

$$\boldsymbol{D}^{(i)}(z) = \mathrm{diag}\{1 \ \ldots \ 1 \ z^{-\tau} \ 1 \ \ldots \ 1\}$$

- $\mathbf{Q}^{(i)}(z)$ is a Givens rotation.

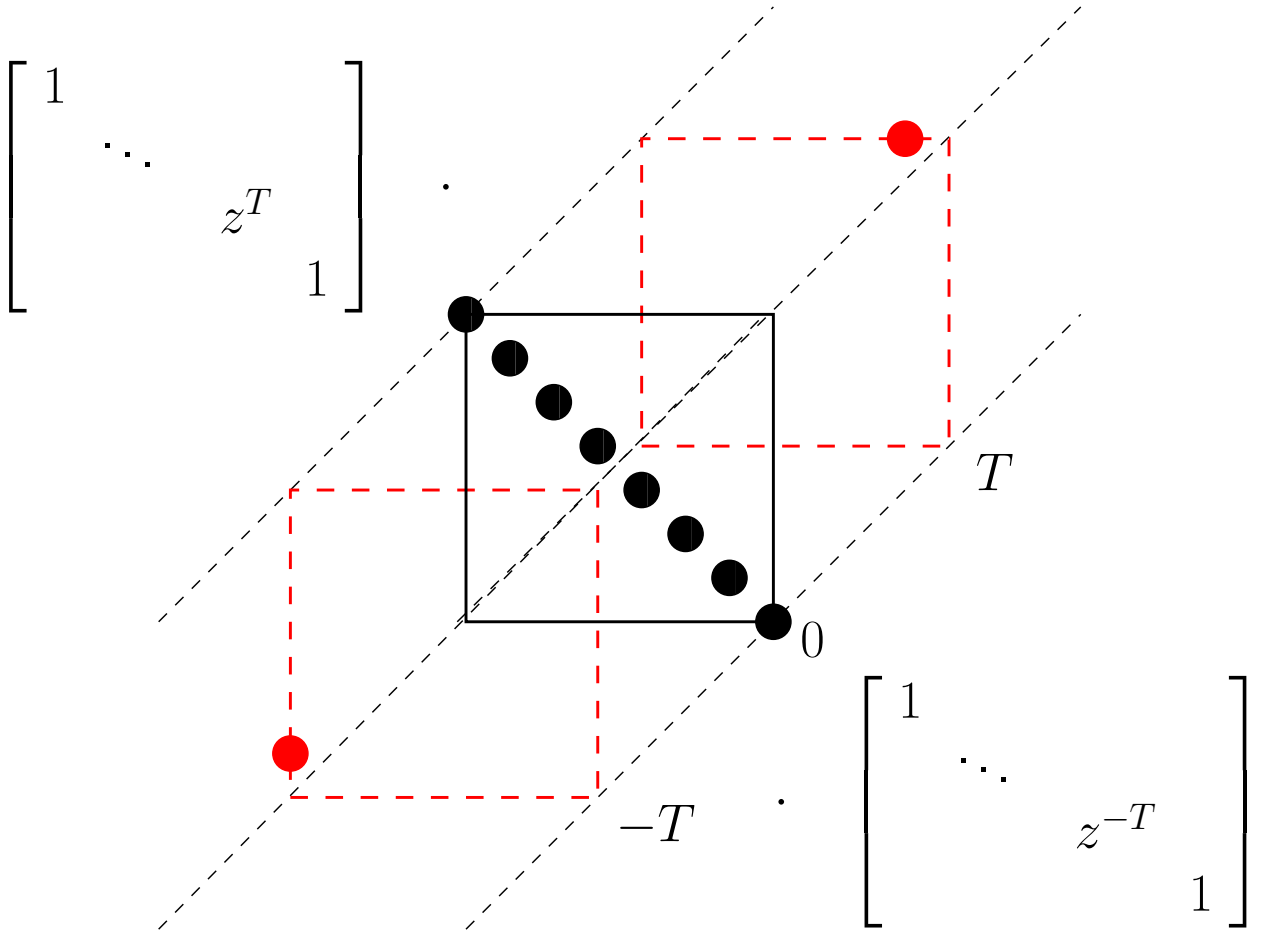
Sequential Best Rotation Algorithm (McWhirter)

- At iteration i, consider $\boldsymbol{S}^{(i-1)}(z) \circ\!\!-\!\!\bullet\, \mathbf{S}^{(i-1)}[\tau]$



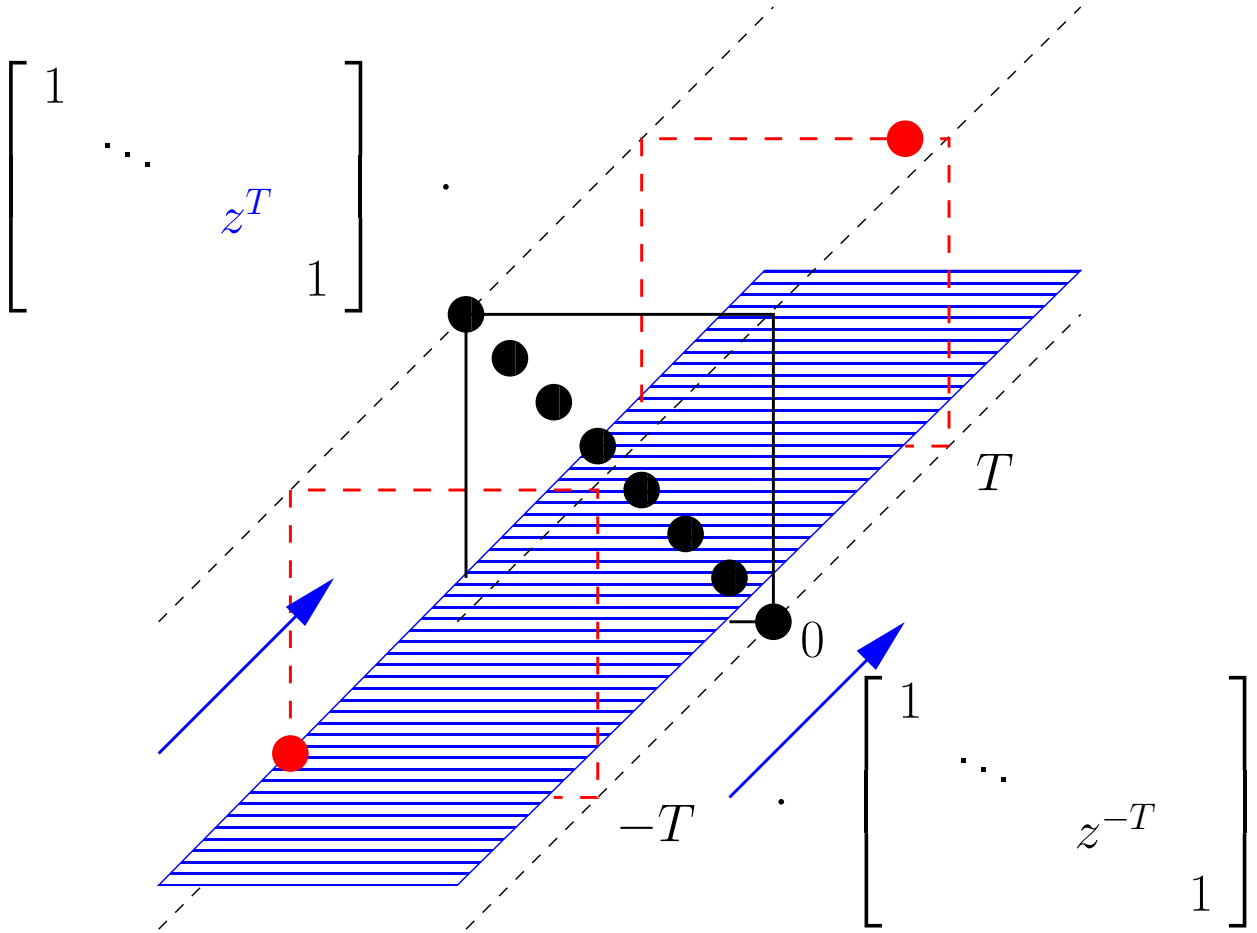
Sequential Best Rotation Algorithm (McWhirter)

- $\tilde{\boldsymbol{D}}^{(i)}(z)\boldsymbol{S}^{(i-1)}(z)\boldsymbol{D}^{(i)}(z)$



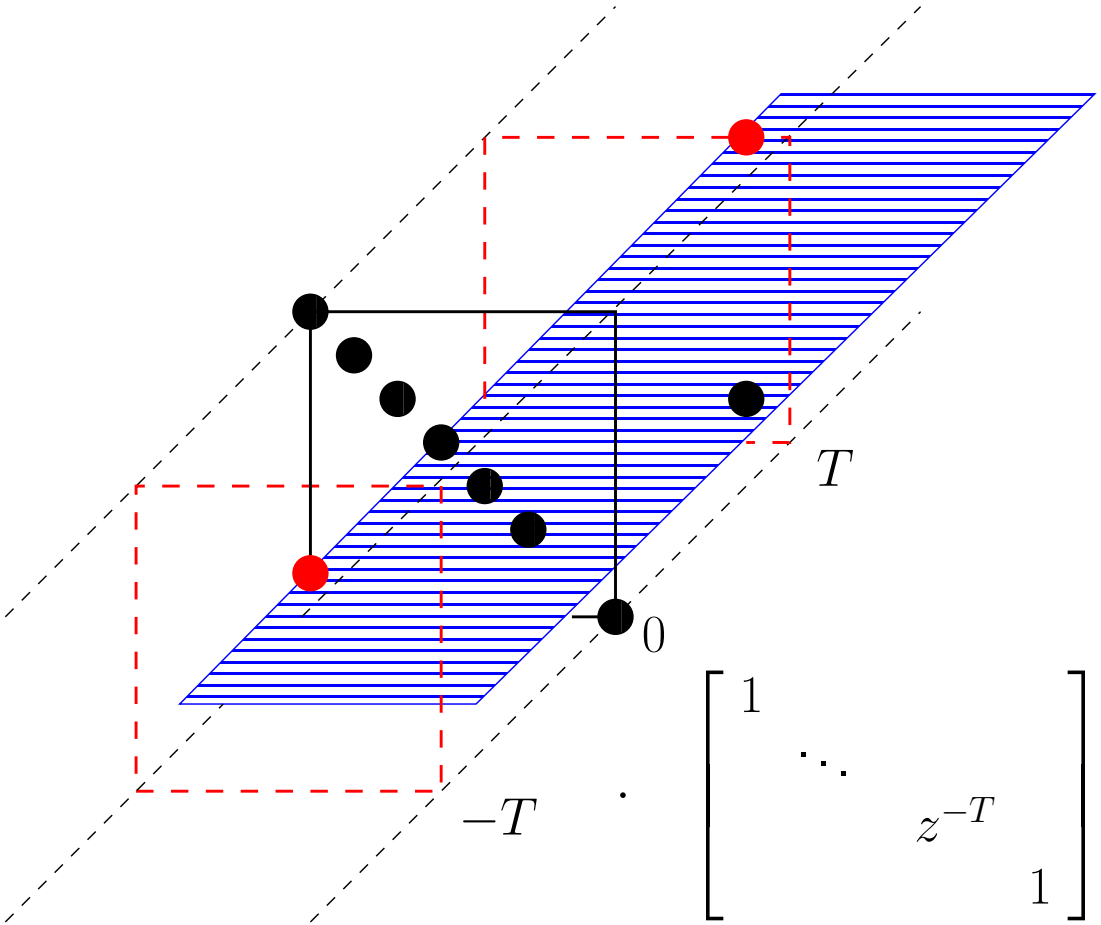
Sequential Best Rotation Algorithm (McWhirter)

- $\tilde{\boldsymbol{D}}^{(i)}(z)$ advances a row-slice of $\boldsymbol{S}^{(i-1)}(z)$ by T



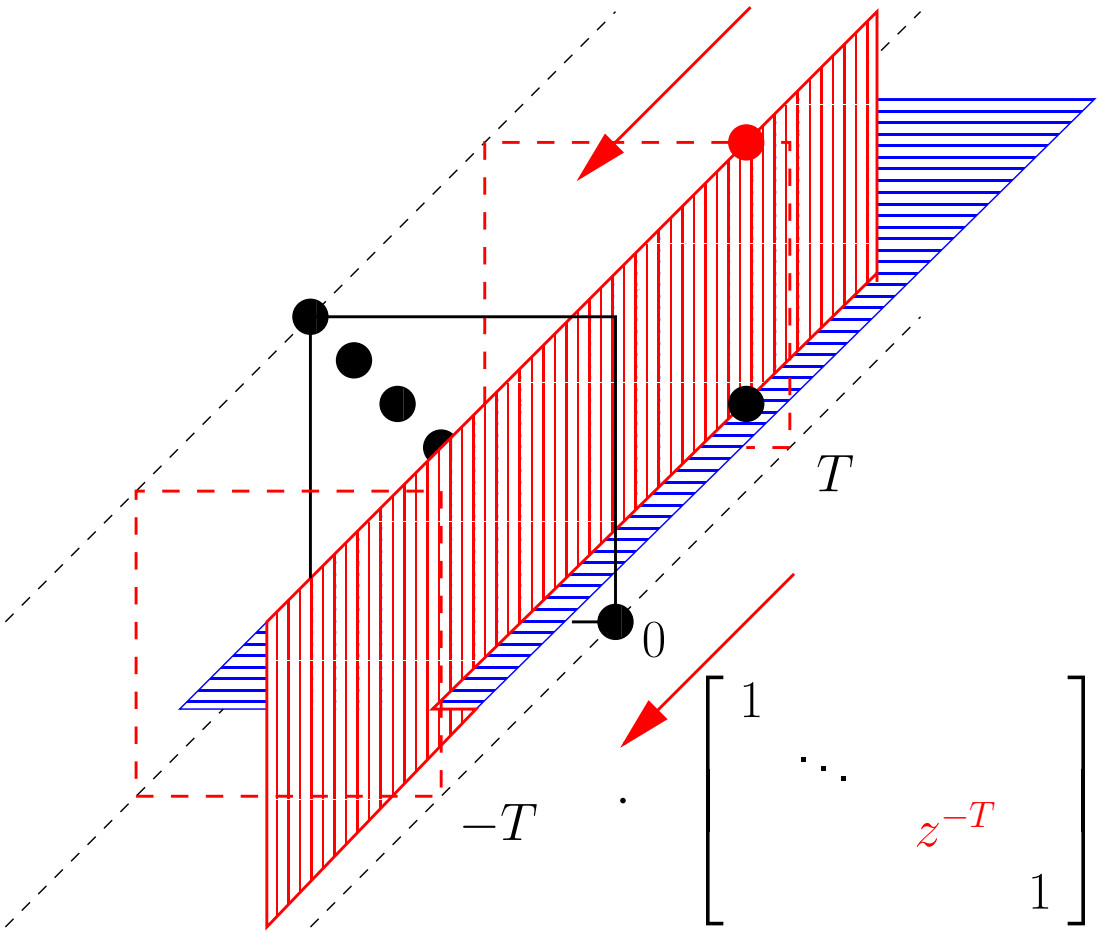
Sequential Best Rotation Algorithm (McWhirter)

- the off-diagonal element at $-T$ has now been translated to lag zero



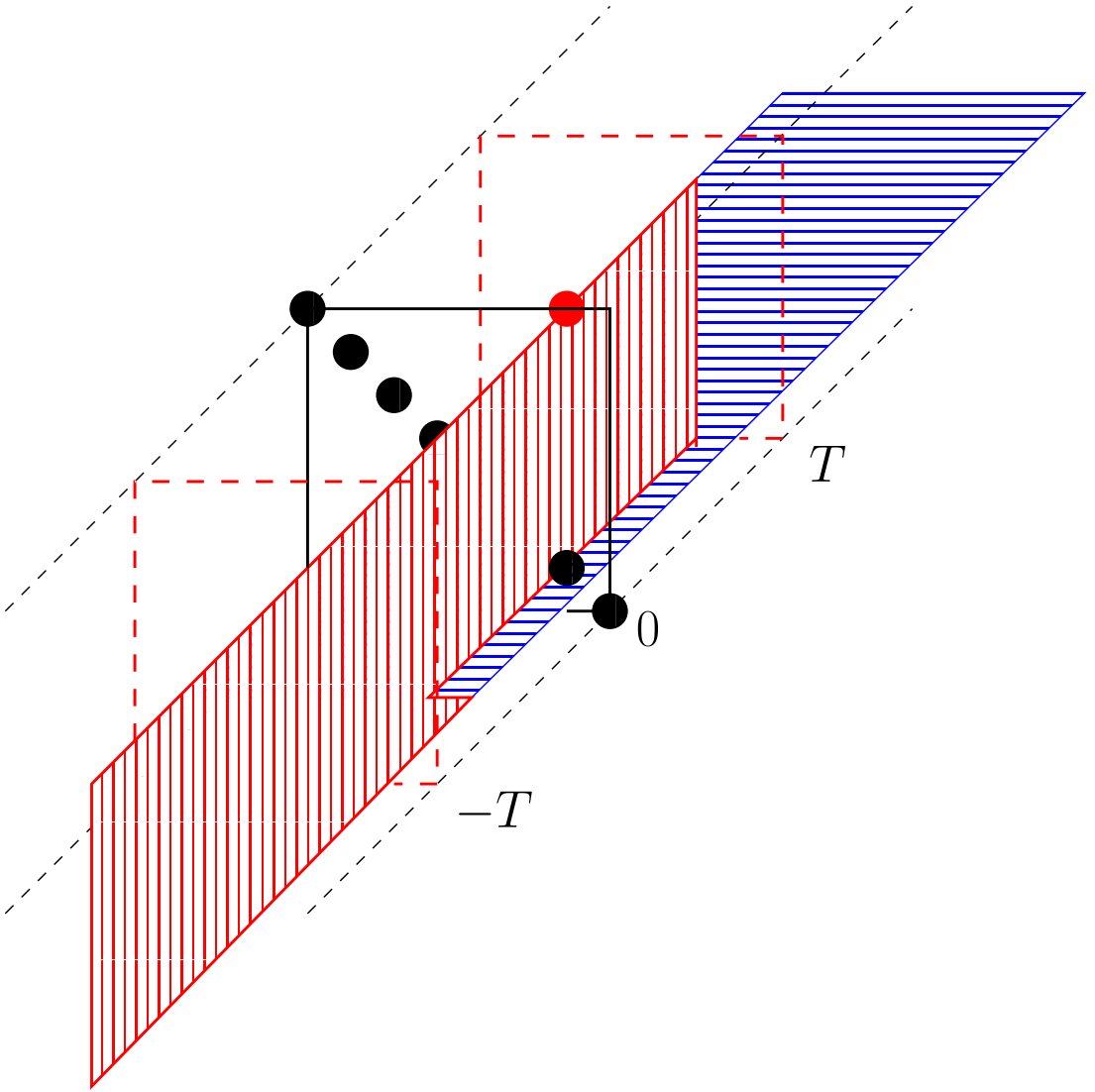
Sequential Best Rotation Algorithm (McWhirter)

- $\mathbf{D}^{(i)}(z)$ delays a column-slice of $\boldsymbol{S}^{(i-1)}(z)$ by T



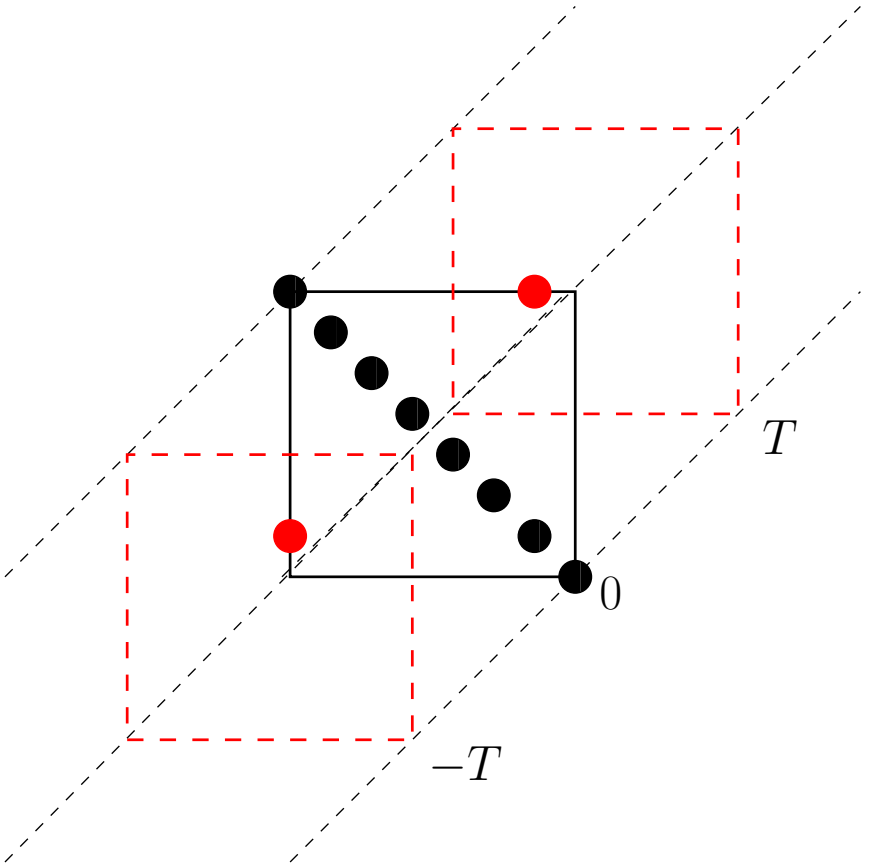
Sequential Best Rotation Algorithm (McWhirter)

- the off-diagonal element at $-T$ has now been translated to lag zero



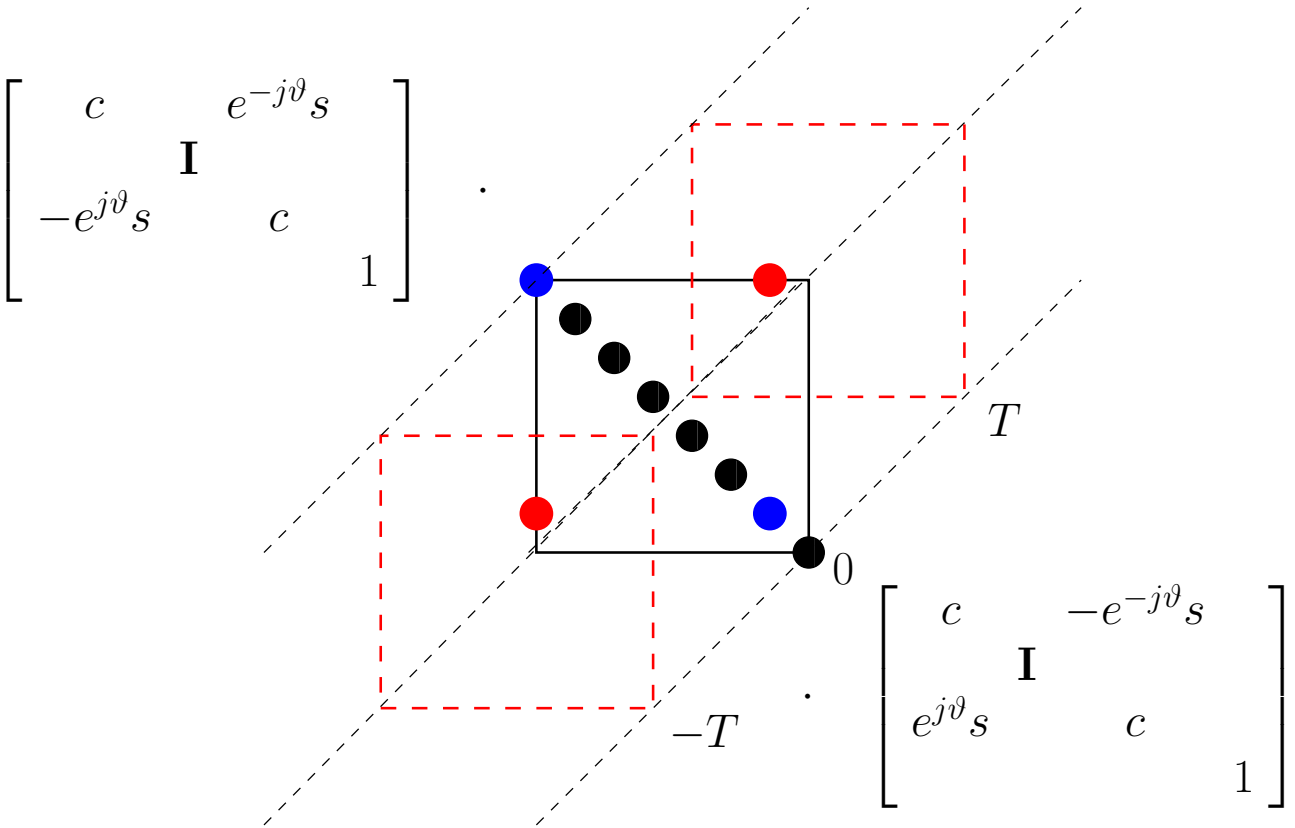
Sequential Best Rotation Algorithm (McWhirter)

- the step $\tilde{\boldsymbol{D}}^{(i)}(z)\boldsymbol{S}^{(i-1)}(z)\boldsymbol{D}_{(i)}(z)$ has brought the largest off-diagonal elements to lag 0.



Sequential Best Rotation Algorithm (McWhirter)

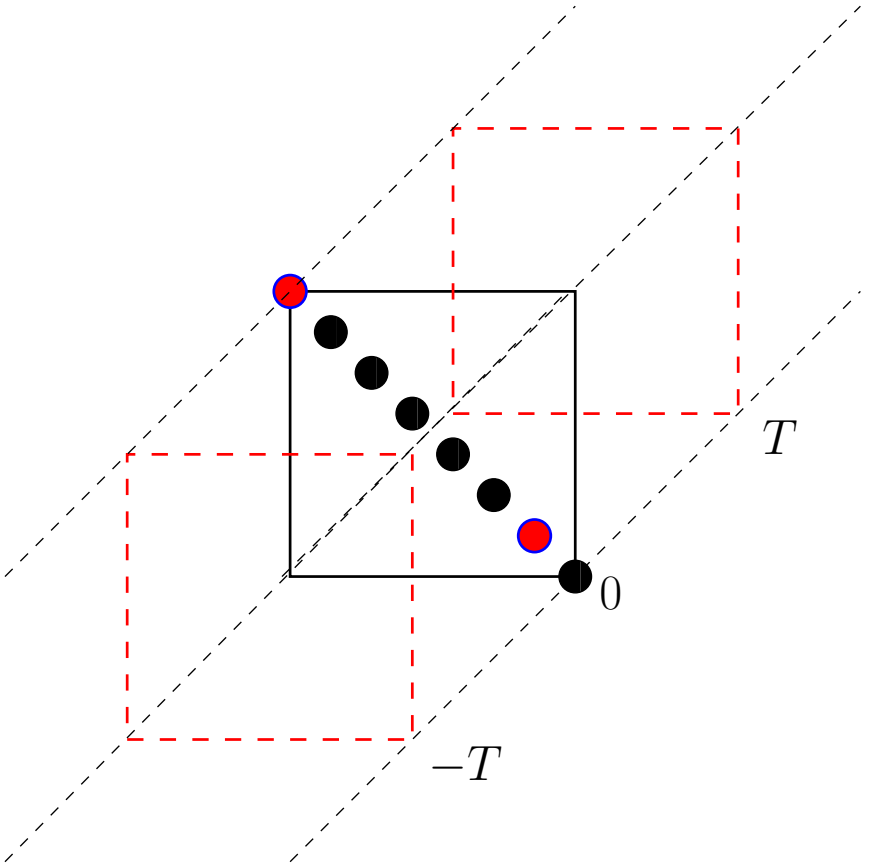
- Jacobi step to eliminate largest off-diagonal elements by $\mathbf{Q}^{(i)}$



Sequential Best Rotation Algorithm (McWhirter)

- iteration i is completed, having performed

$$\boldsymbol{S}^{(i)}(z) = \mathbf{Q}^{(i)}\boldsymbol{D}^{(i)}(z)\boldsymbol{S}^{(i-1)}(z)\tilde{\boldsymbol{D}}^{(i)}(z)\tilde{\mathbf{Q}}^{(i)}(z)$$



SBR2 Outcome

- At the ith iteration, the zeroing of off-diagonal elements achieved during previous steps may be partially undone;
- however, the algorithm has been shown to converge, transfering energy onto the main diagonal at every step (McWhirter 2007);
- after L iterations, we reach an approximate diagonalisation

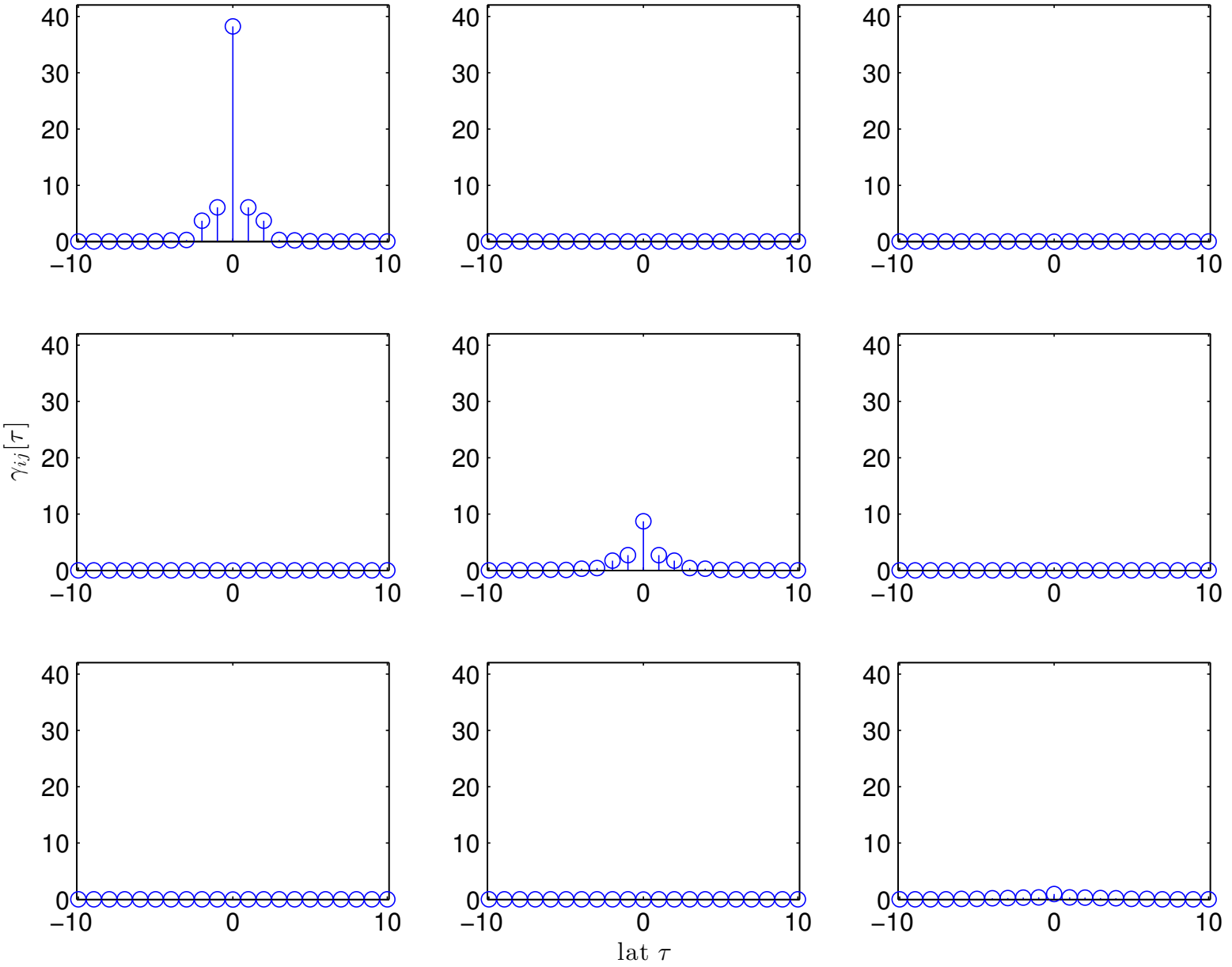
$$\hat{\mathbf{\Lambda}}(z) = \boldsymbol{S}^{(L)}(z) = \tilde{\boldsymbol{Q}}(z)\boldsymbol{R}(z)\boldsymbol{Q}(z)$$

with

$$\boldsymbol{Q}(z) = \prod_{i=1}^{L} \boldsymbol{D}^{(i)}(z)\mathbf{Q}^{(i)}$$

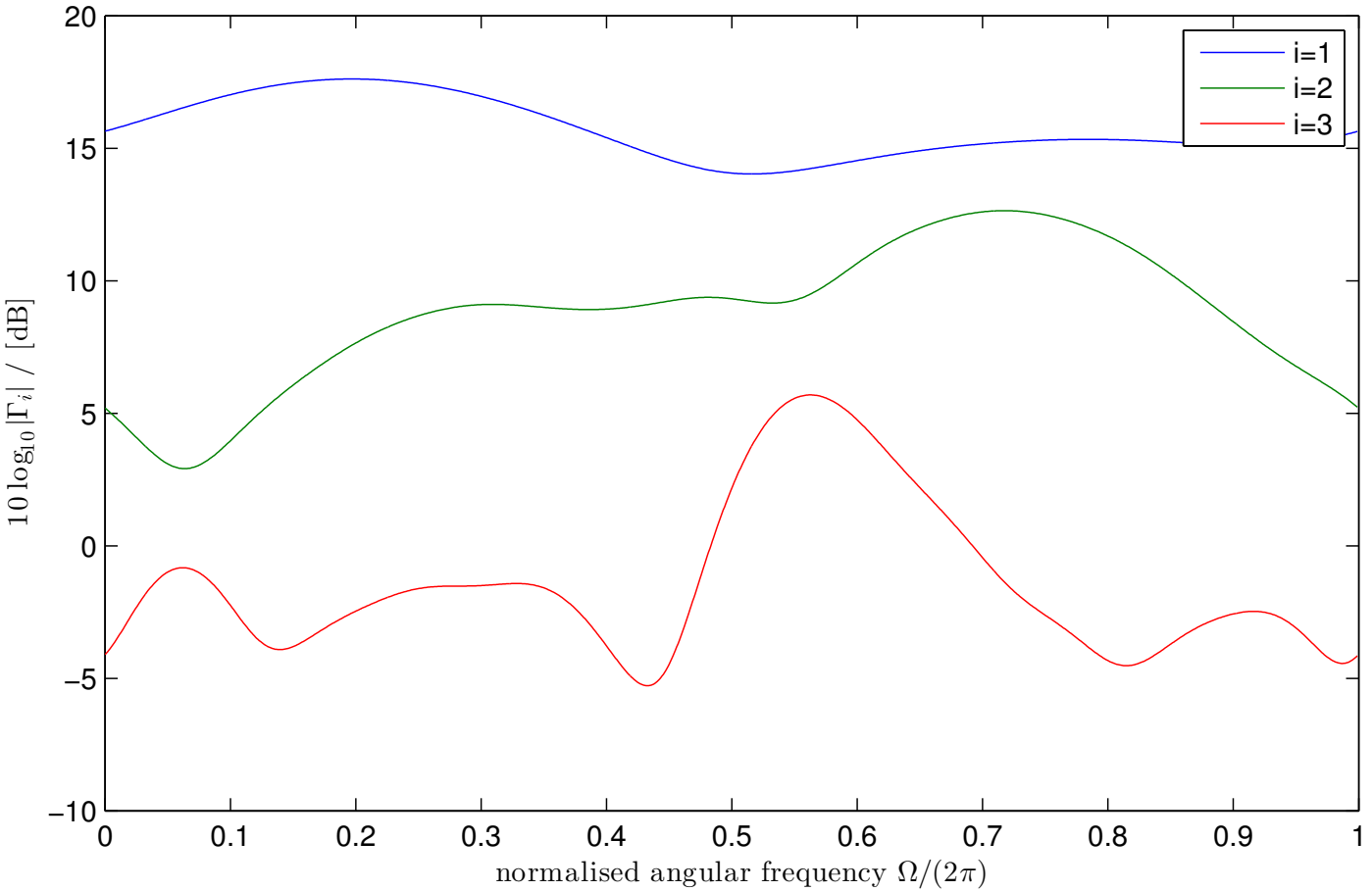
- diagonalisation of the previous 3×3 polynomial matrix ...

SBR2 Example — Diagonalisation



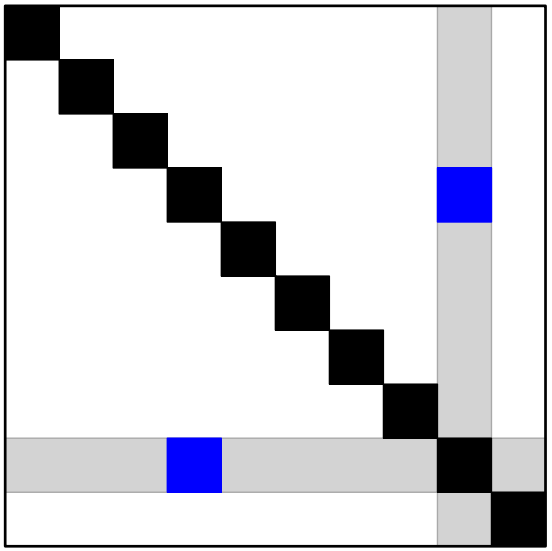
SBR2 Example — Spectral Majorisation

- The on-diagonal elements are spectrally majorised



SBR2 — Givens Rotation

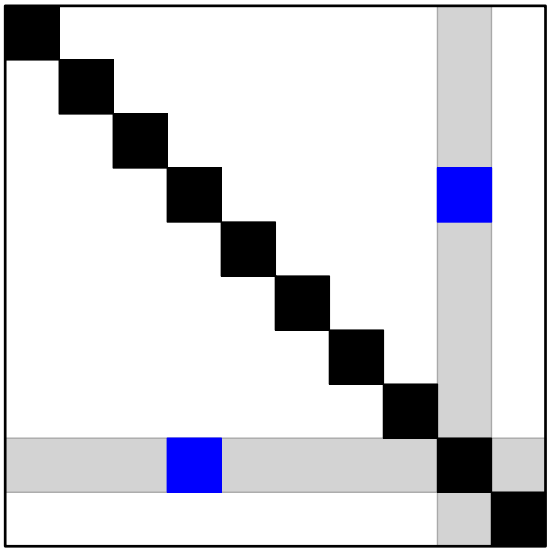
- A Givens rotation eliminates the maximum off-diagonal element once brought onto the lag-zero matrix;
- note I: in the lag-zero matrix, one column and one row are modified by the shift:

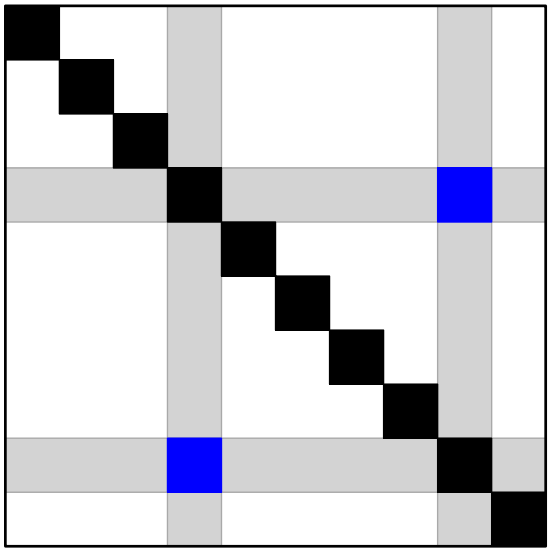


- note II: a Givens rotation only affects two columns and two rows in every matrix;
- Givens rotation is relatively low in computational cost!

SBR2 — Givens Rotation

- A Givens rotation eliminates the maximum off-diagonal element once brought onto the lag-zero matrix;
- note I: in the lag-zero matrix, one column and one row are modified by the shift:



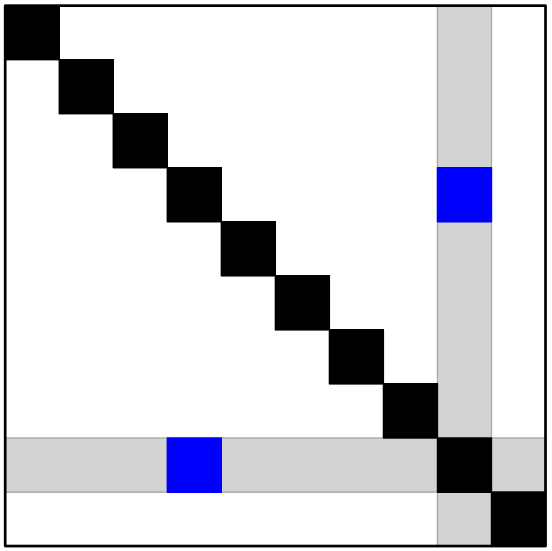


- note II: a Givens rotation only affects two columns and two rows in every matrix;
- Givens rotation is relatively low in computational cost!

Sequential Matrix Diagonalisation (SMD)

[Redif *et al.*, IEEE Trans SP 2015]

- Main idea — the zero-lag matrix is diagonalised in every step;
- initialisation: diagonalise $\mathbf{R}[0]$ by EVD and apply modal matrix to all matrix coefficients $\longrightarrow \boldsymbol{S}^{(0)}$;
- at the ith step as in SBR2, the maximum element (or column with max. norm) is shifted to the lag-zero matrix:

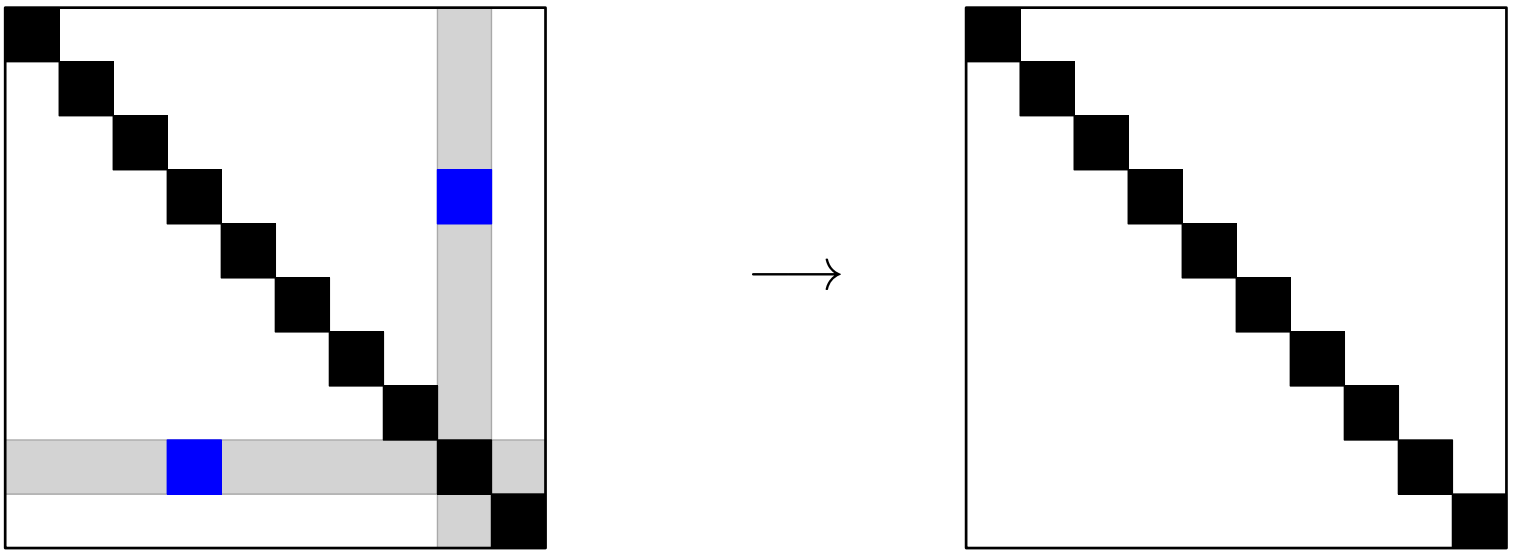


- an EVD is used to re-diagonalise the zero-lag matrix;
- a full modal matrix is applied at all lags — more costly than SBR2.

Sequential Matrix Diagonalisation (SMD)

[Redif *et al.*, IEEE Trans SP 2015]

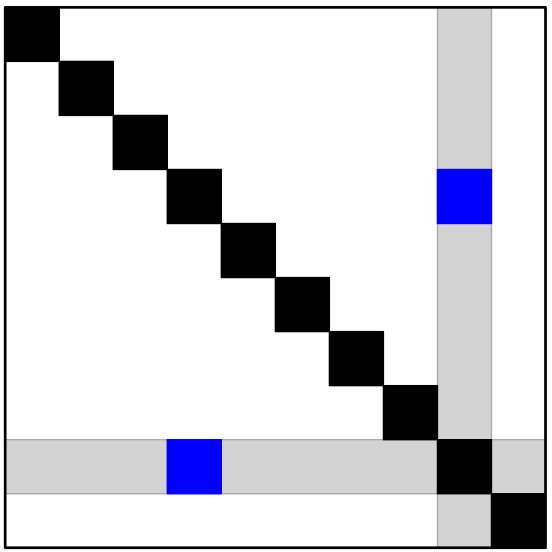
- Main idea — the zero-lag matrix is diagonalised in every step;
- initialisation: diagonalise $\mathbf{R}[0]$ by EVD and apply modal matrix to all matrix coefficients $\longrightarrow \boldsymbol{S}^{(0)}$;
- at the ith step as in SBR2, the maximum element (or column with max. norm) is shifted to the lag-zero matrix:



- an EVD is used to re-diagonalise the zero-lag matrix;
- a full modal matrix is applied at all lags — more costly than SBR2.

Multiple Shift SMD (SMD)

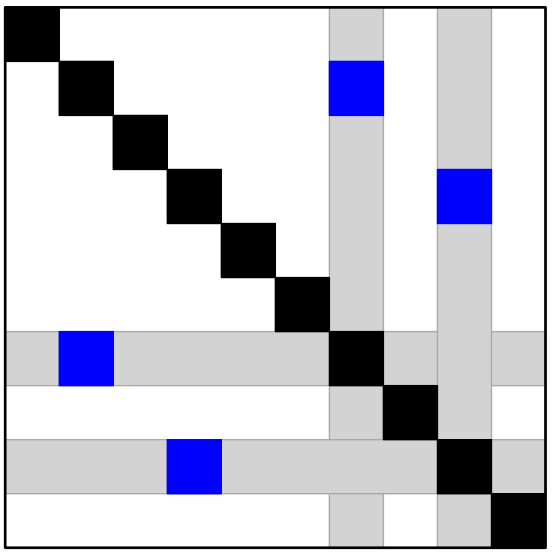
- SMD converges faster than SBR2 — more energy is transfered per iteration step;
- SMD is more expensive than SBR2 — full matrix multiplication at every lag;
- this cost will not increase further if more columns / rows are shifted into the lag-zero matrix at every iteration



- MS-SMD will transfer yet more off-diagonal energy per iteration;
- because the total energy must remain constant under paraunitary operations, SBR2, SMD and MS-SMD can be proven to converge.

Multiple Shift SMD (SMD)

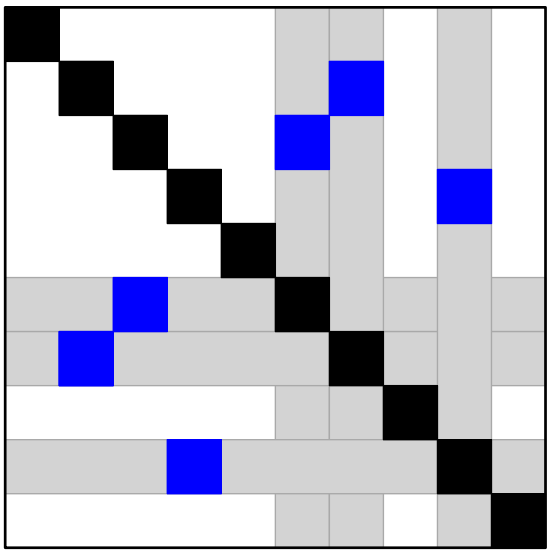
- SMD converges faster than SBR2 — more energy is transfered per iteration step;
- SMD is more expensive than SBR2 — full matrix multiplication at every lag;
- this cost will not increase further if more columns / rows are shifted into the lag-zero matrix at every iteration



- MS-SMD will transfer yet more off-diagonal energy per iteration;
- because the total energy must remain constant under paraunitary operations, SBR2, SMD and MS-SMD can be proven to converge.

Multiple Shift SMD (SMD)

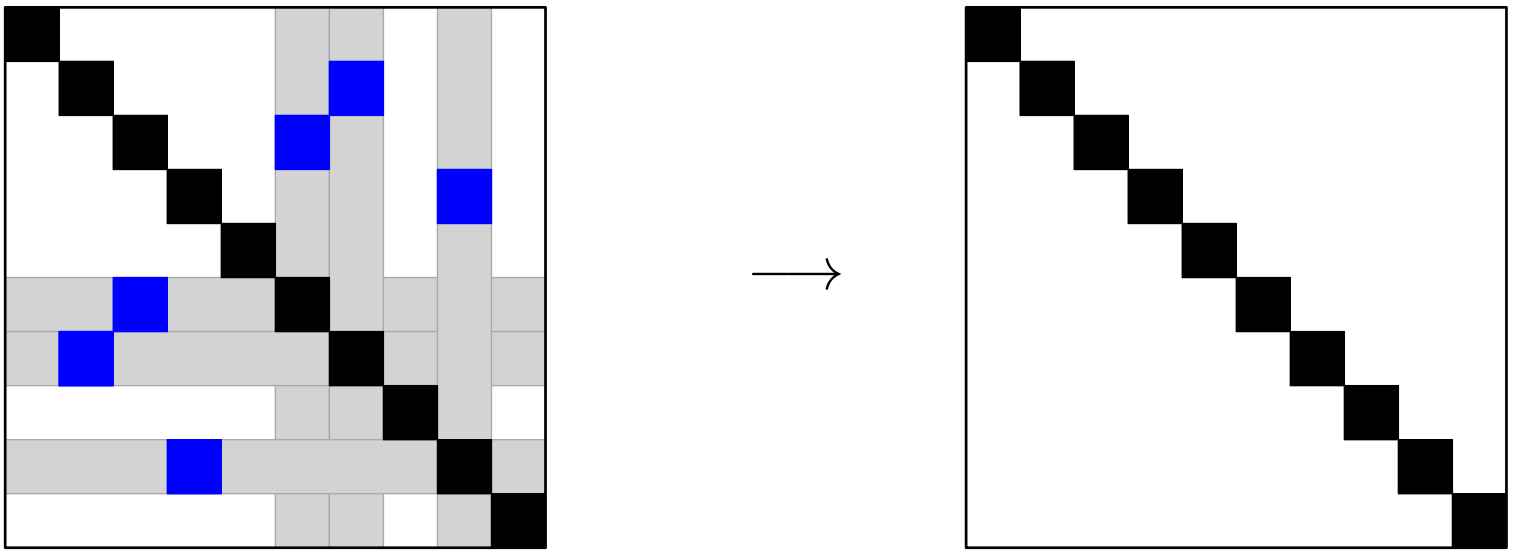
- SMD converges faster than SBR2 — more energy is transfered per iteration step;
- SMD is more expensive than SBR2 — full matrix multiplication at every lag;
- this cost will not increase further if more columns / rows are shifted into the lag-zero matrix at every iteration



- MS-SMD will transfer yet more off-diagonal energy per iteration;
- because the total energy must remain constant under paraunitary operations, SBR2, SMD and MS-SMD can be proven to converge.

Multiple Shift SMD (SMD)

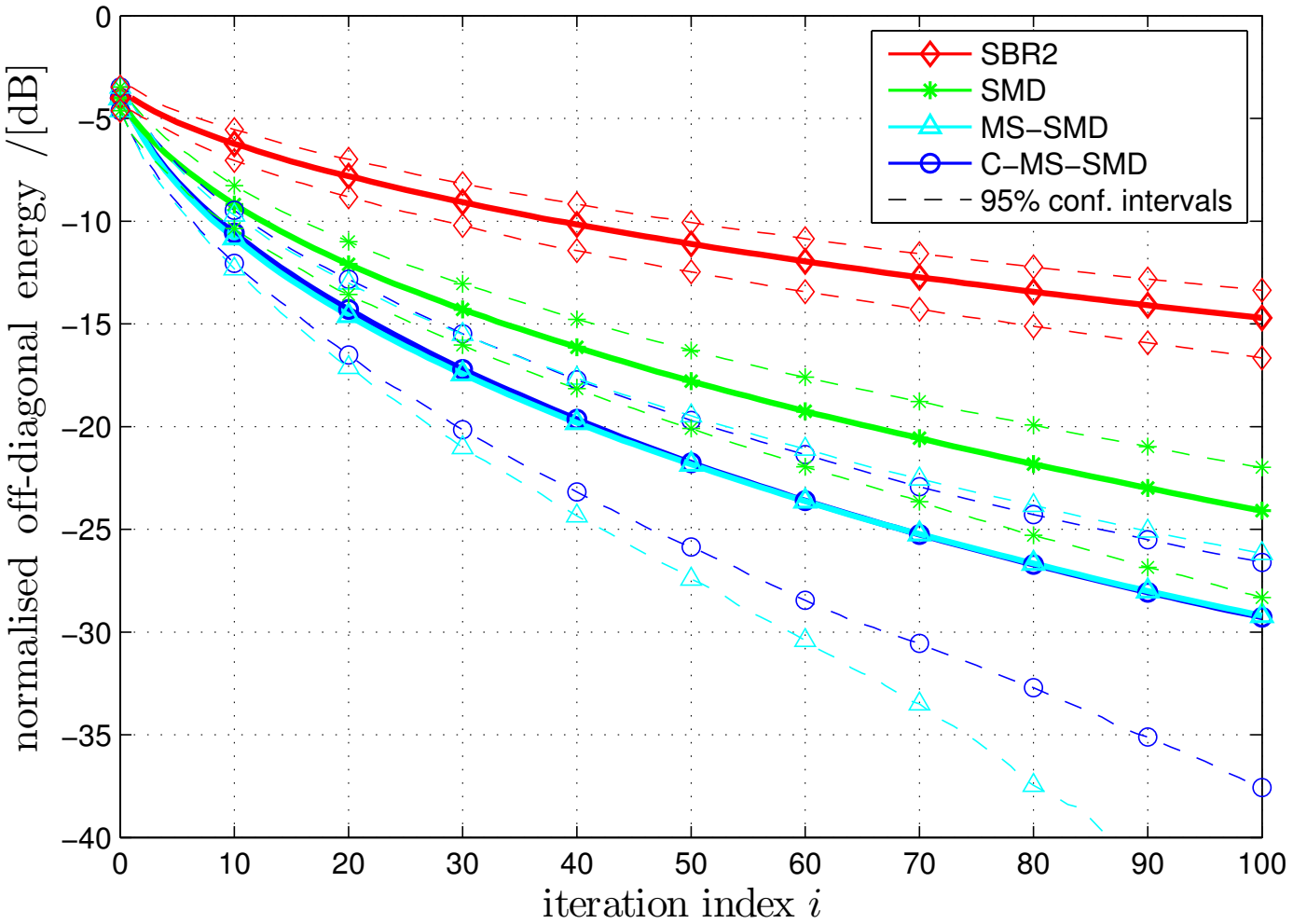
- SMD converges faster than SBR2 — more energy is transfered per iteration step;
- SMD is more expensive than SBR2 — full matrix multiplication at every lag;
- this cost will not increase further if more columns / rows are shifted into the lag-zero matrix at every iteration



- MS-SMD will transfer yet more off-diagonal energy per iteration;
- because the total energy must remain constant under paraunitary operations, SBR2, SMD and MS-SMD can be proven to converge.

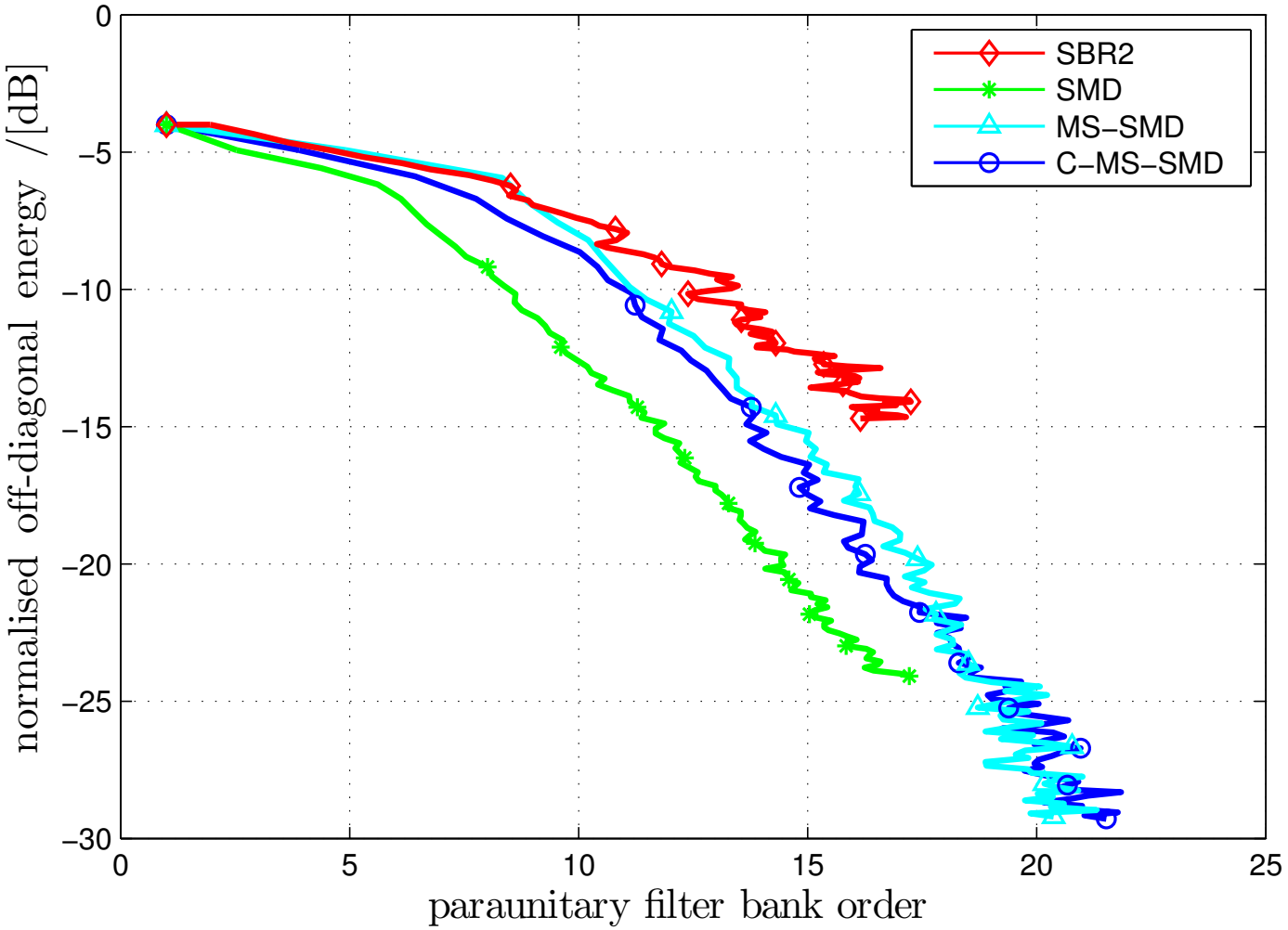
SBR2/SMD/MS-SMD Convergence

- Measuring the remaining normalised off-diagonal energy over an ensemble of space-time covariance matrices:



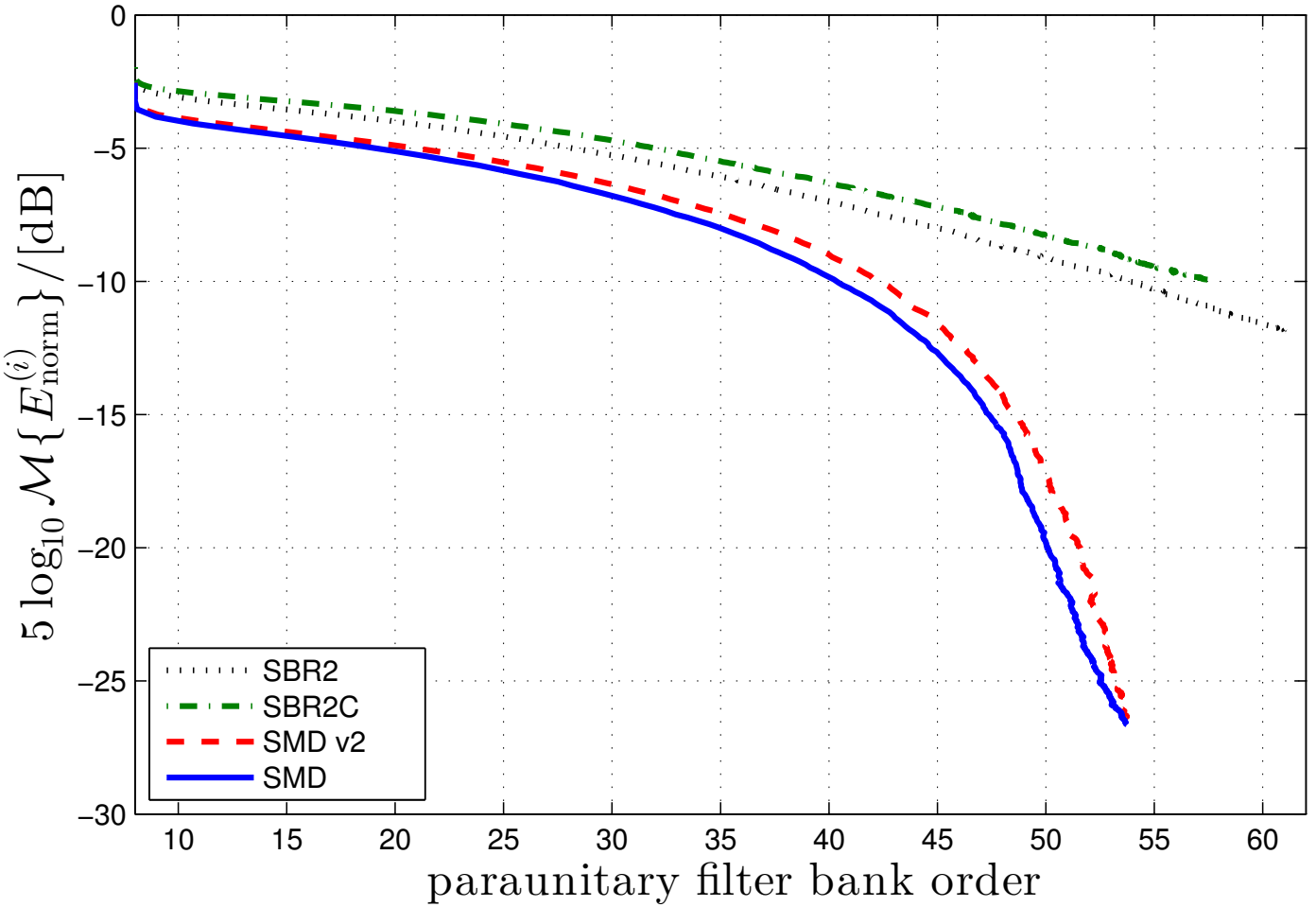
SBR2/SMD/MS-SMD Application Cost 1

- Ensemble average of remaining off-diagonal energy vs. order of paraunitary filter banks to decompose 4x4x16 matrices:



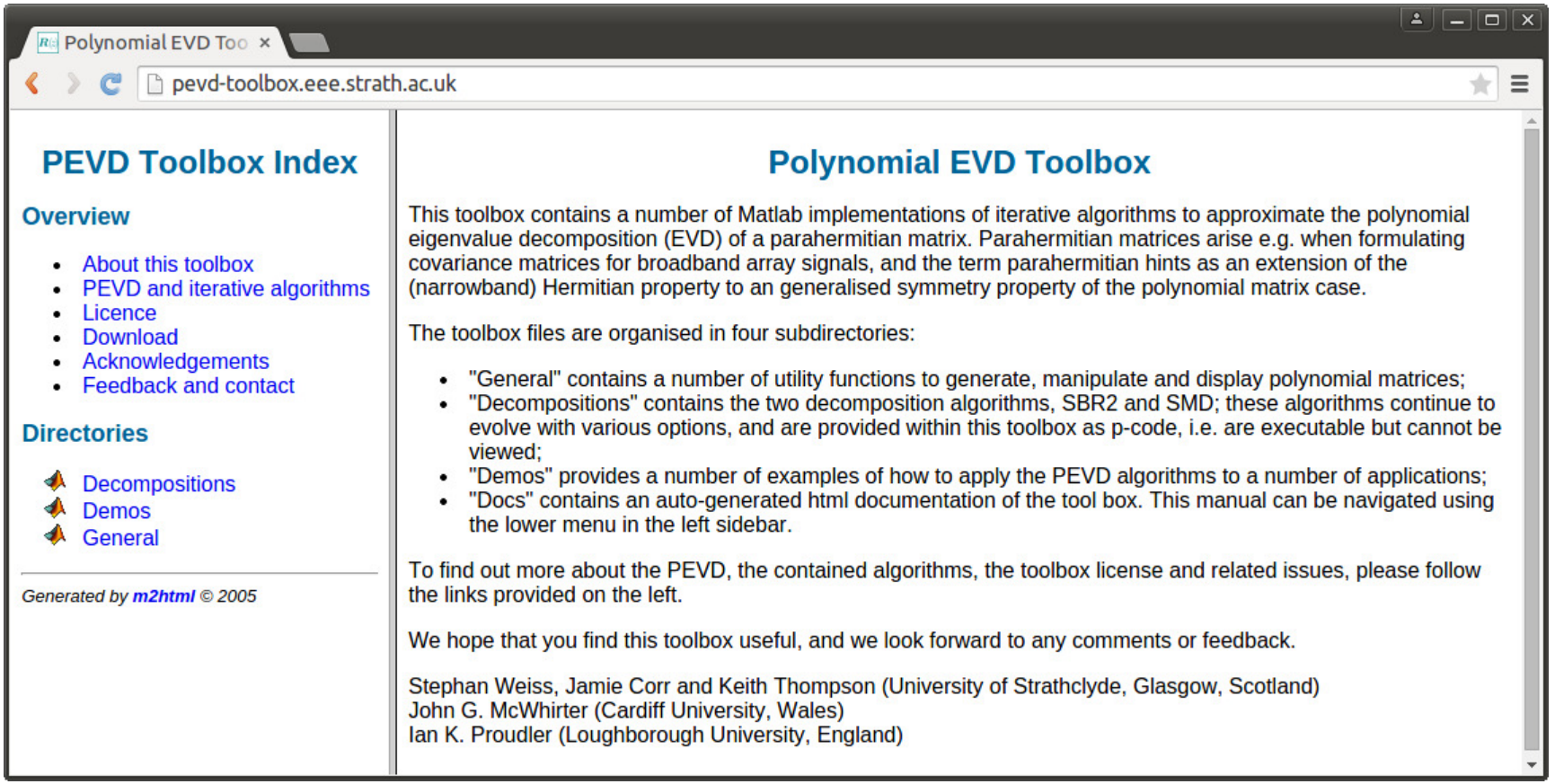
SBR2/SMD/MS-SMD Application Cost 2

- Ensemble average of remaining off-diagonal energy vs. order of paraunitary filter banks to decompose 8x8x64 matrices:



MATLAB Polynomial EVD Toolbox

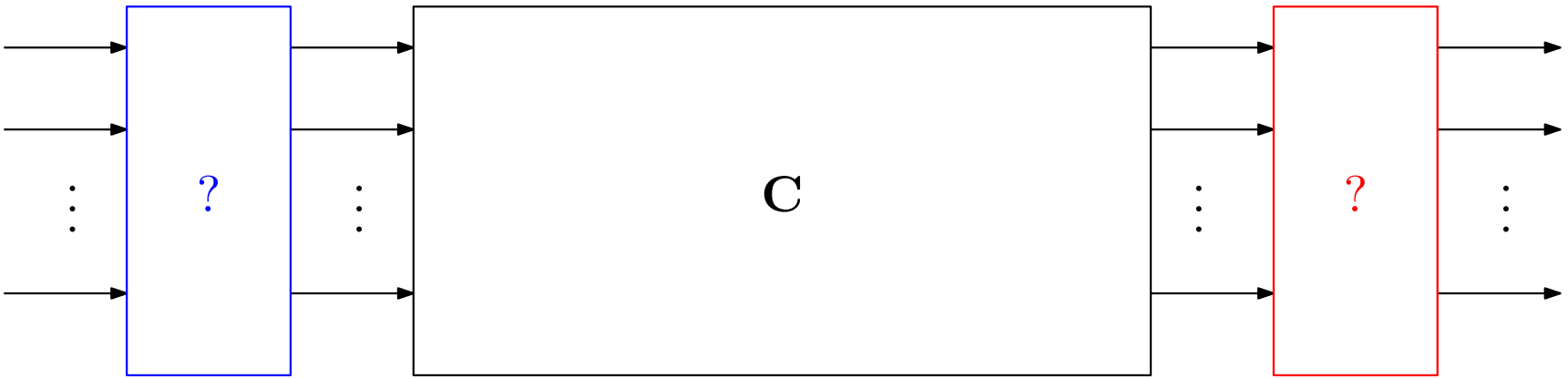
- The MATLAB polynomial EVD toolbox can be downloaded from
  `pevd-toolbox.eee.strath.ac.uk`



- the toolbox contains a number of iterative algorithms to calculate an approximate PEVD, related functions, and demos.

Narrowband MIMO Communications

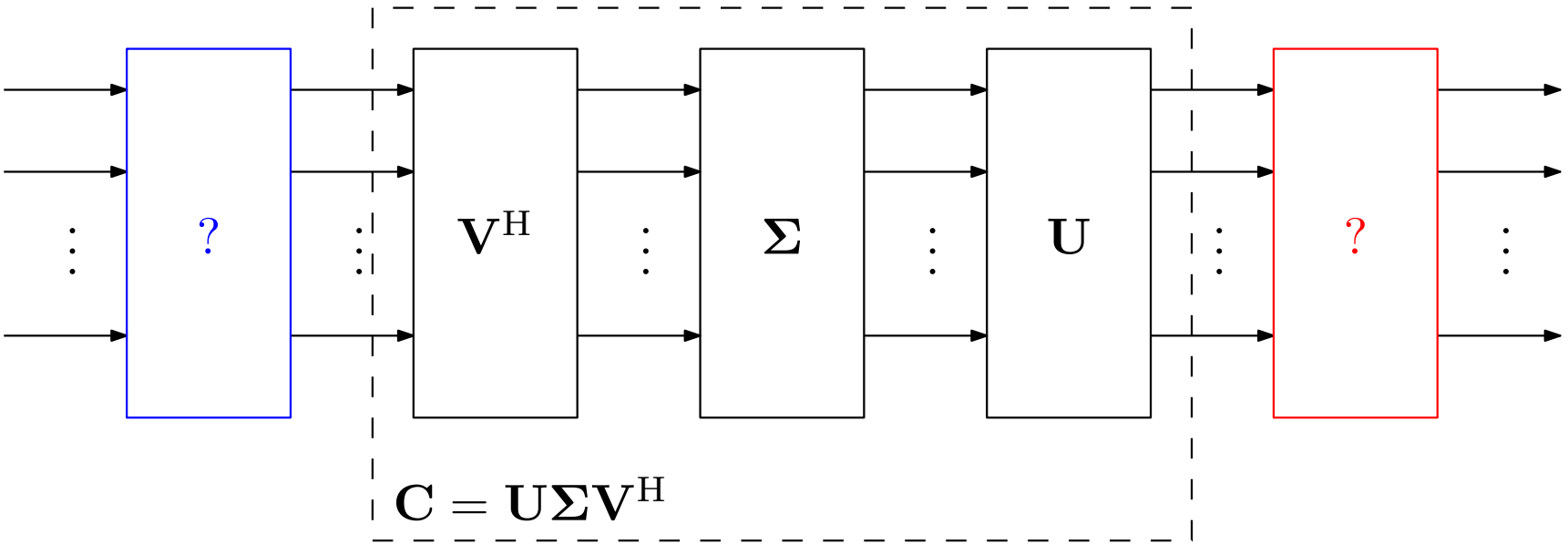
- a narrowband channel is characterised by a matrix $\mathbf{C}$ containing complex gain factors;
- problem: how to select the precoder and equaliser?



- overall system;

Narrowband MIMO Communications

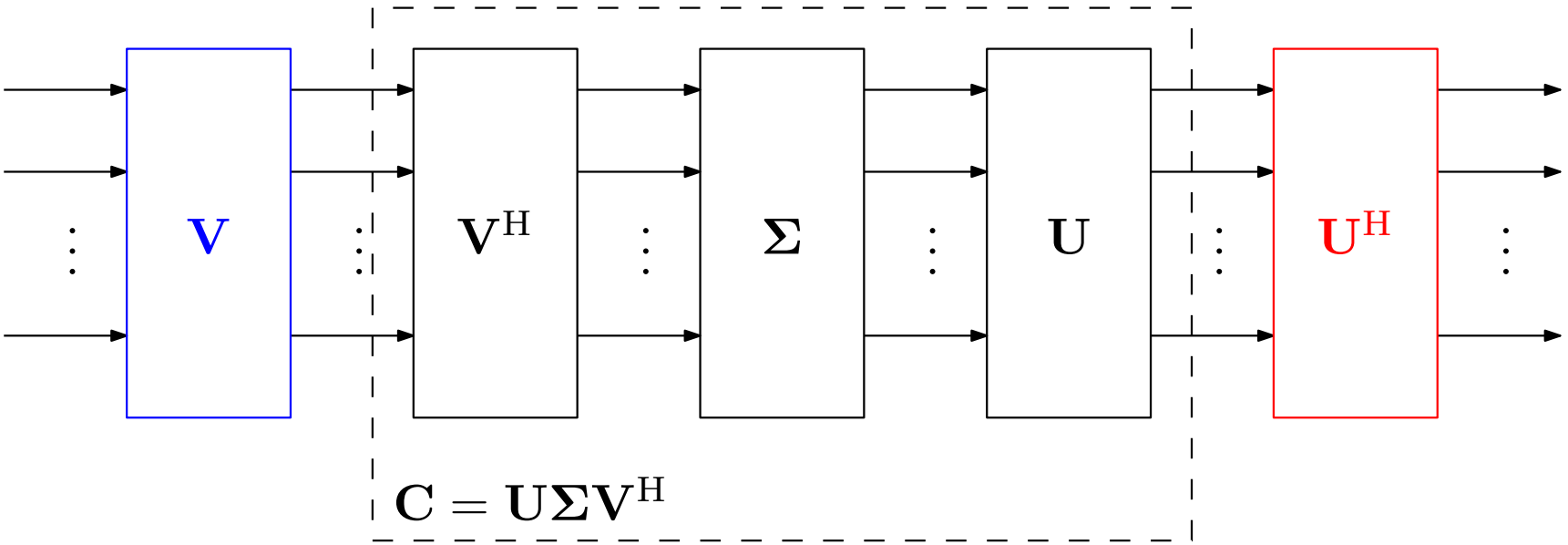
- a narrowband channel is characterised by a matrix $\mathbf{C}$ containing complex gain factors;
- problem: how to select the precoder and equaliser?



- the singular value decomposition (SVD) factorises $\mathbf{C}$ into two unitary matrices U and $\mathbf{V}^{\mathrm{H}}$ and a diagonal matrix $\mathbf{\Sigma}$;

Narrowband MIMO Communications

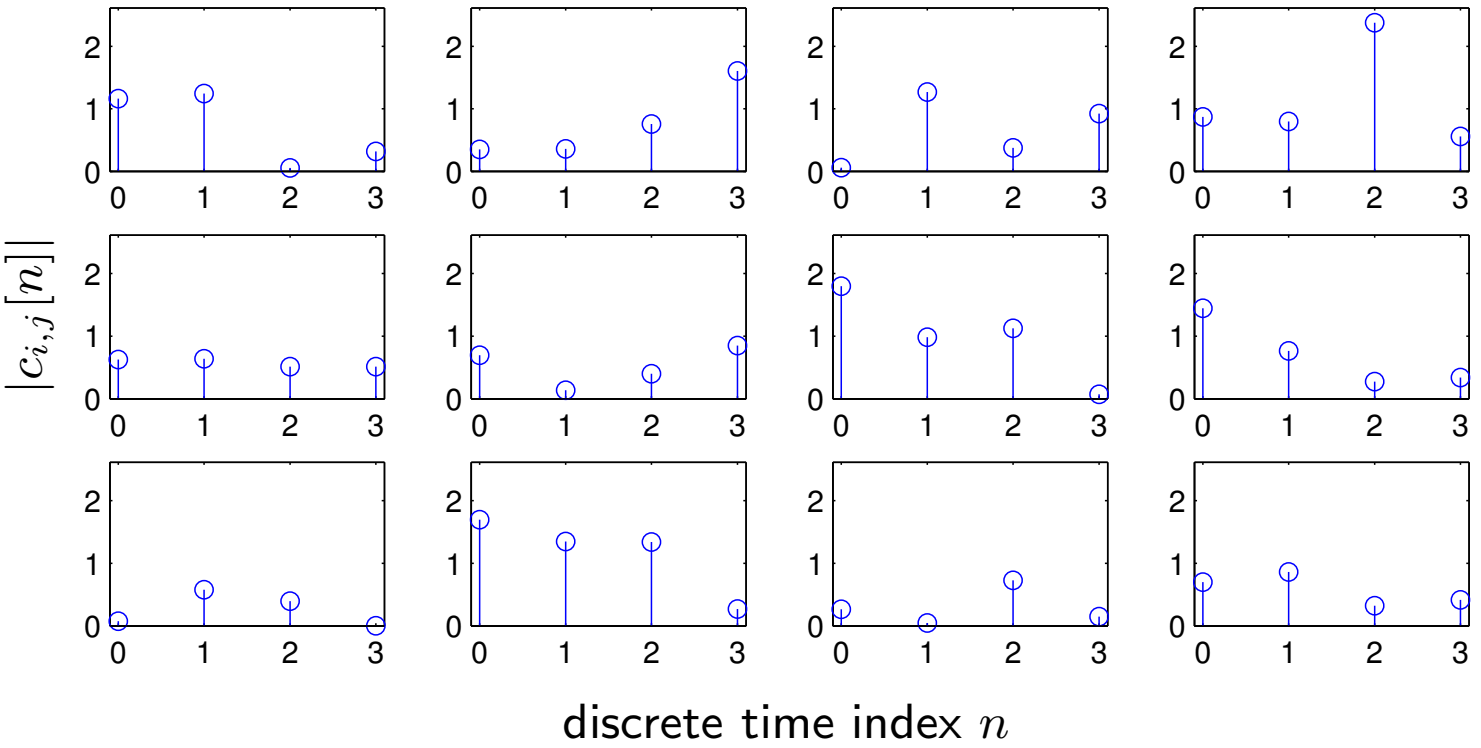
- a narrowband channel is characterised by a matrix $\mathbf{C}$ containing complex gain factors;
- problem: how to select the precoder and equaliser?



- we select the precoder and the equaliser from the unitary matrices provided by the channel's SVD;
- the overall system is diagonalised, decoupling the channel into independent single-input single-output systems by means of unitary matrices.

Broadband MIMO Channel

- The channel is now a matrix of FIR filters; example for a 3×4 MIMO system $\mathbf{C}[n]$:

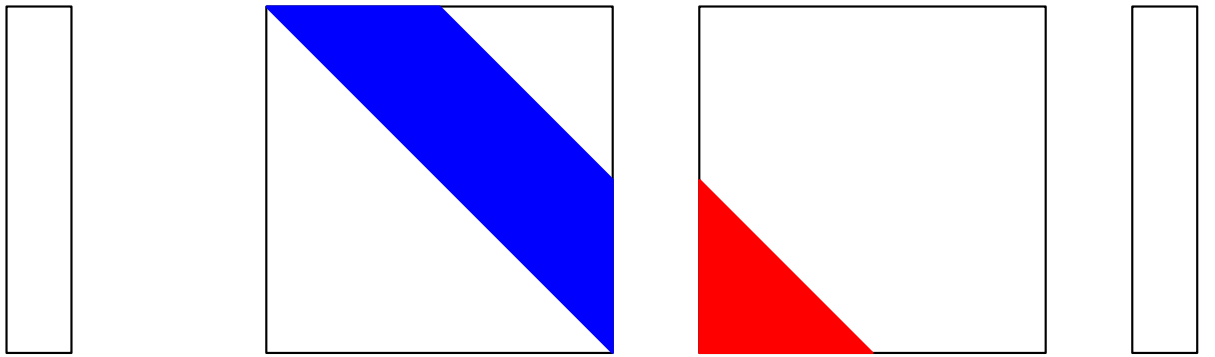


- the transfer function $\boldsymbol{C}(z) \bullet\!\!-\!\!\circ\, \mathbf{C}[n]$ is a polynomial matrix;
- an SVD can only diagonalise $\mathbf{C}[n]$ for one particular lag n.

Standard Broadband MIMO Approaches

- OFDM (if approximate channel length is known):
  1. divide spectrum into narrowband channels;
  2. address each narrowband channel independently using narrowband-optimal techniques;

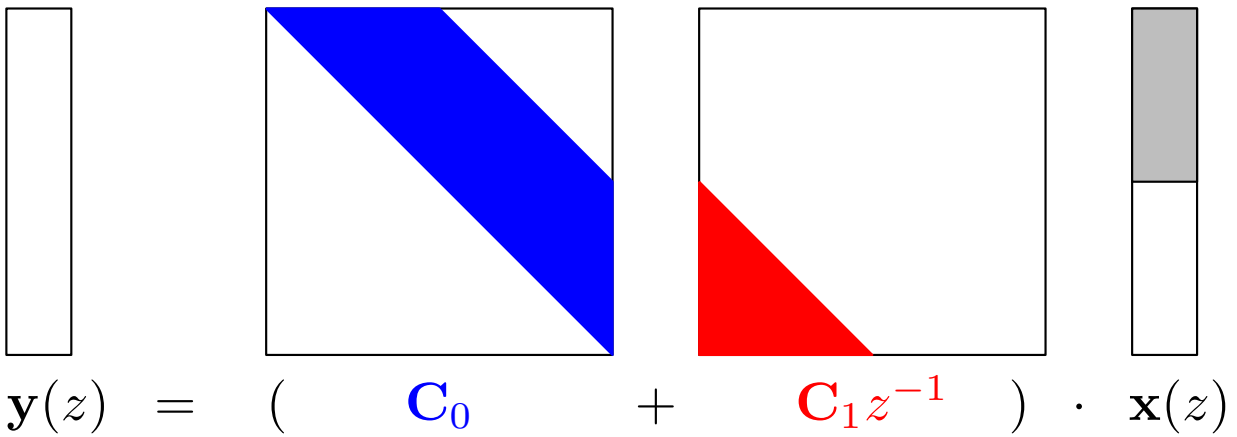
  drawback: ignores spectral coherence across frequency bins;
- optimum filter bank transceiver (if channel itself is known):
  1. block processing;
  2. inter-block interference is eliminated by guard intervals;
  3. resulting matrix can be diagonalised by SVD;
- both techniques invest DOFs into the guard intervals, which are generally not balanced against other error sources.



Standard Broadband MIMO Approaches

- OFDM (if approximate channel length is known):
  1. divide spectrum into narrowband channels;
  2. address each narrowband channel independently using narrowband-optimal techniques;

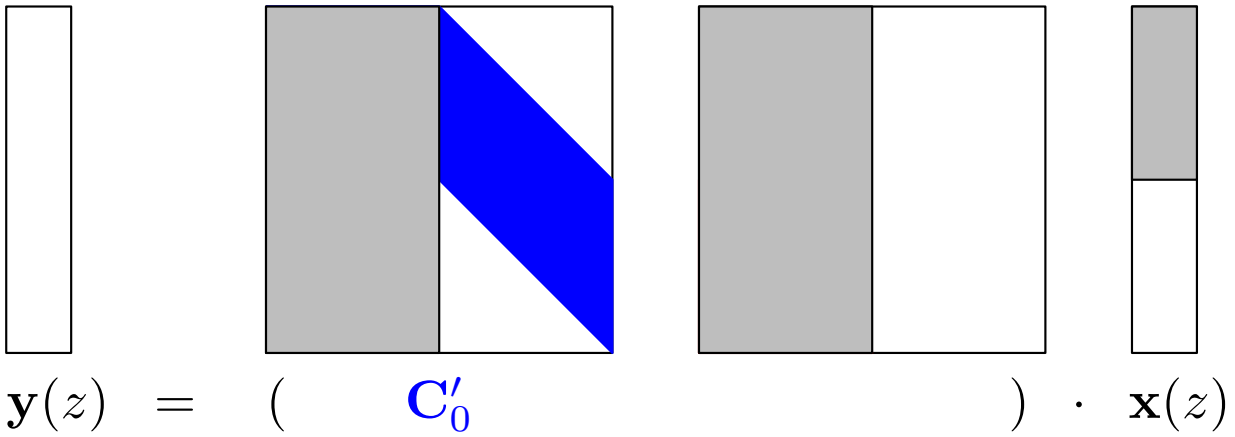
  drawback: ignores spectral coherence across frequency bins;
- optimum filter bank transceiver (if channel itself is known):
  1. block processing;
  2. inter-block interference is eliminated by guard intervals;
  3. resulting matrix can be diagonalised by SVD;
- both techniques invest DOFs into the guard intervals, which are generally not balanced against other error sources.



Standard Broadband MIMO Approaches

- OFDM (if approximate channel length is known):
  1. divide spectrum into narrowband channels;
  2. address each narrowband channel independently using narrowband-optimal techniques;

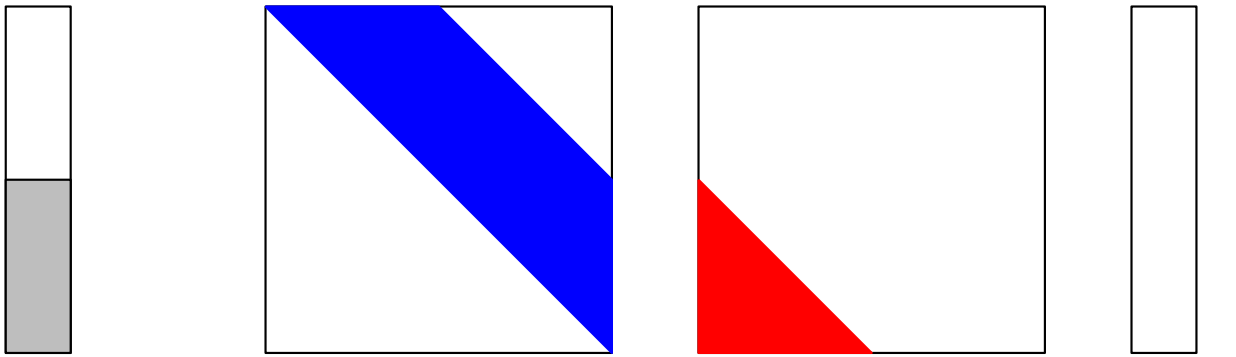
  drawback: ignores spectral coherence across frequency bins;
- optimum filter bank transceiver (if channel itself is known):
  1. block processing;
  2. inter-block interference is eliminated by guard intervals;
  3. resulting matrix can be diagonalised by SVD;
- both techniques invest DOFs into the guard intervals, which are generally not balanced against other error sources.



Standard Broadband MIMO Approaches

- OFDM (if approximate channel length is known):
  1. divide spectrum into narrowband channels;
  2. address each narrowband channel independently using narrowband-optimal techniques;

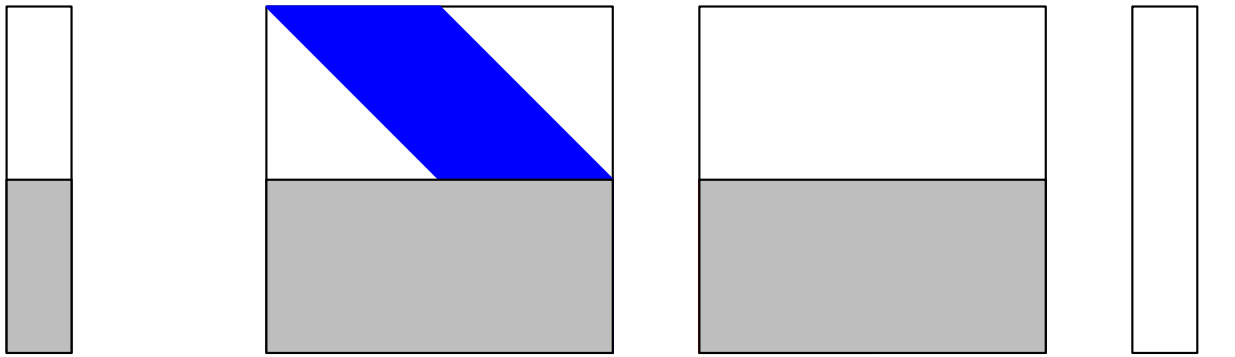
  drawback: ignores spectral coherence across frequency bins;
- optimum filter bank transceiver (if channel itself is known):
  1. block processing;
  2. inter-block interference is eliminated by guard intervals;
  3. resulting matrix can be diagonalised by SVD;
- both techniques invest DOFs into the guard intervals, which are generally not balanced against other error sources.



Standard Broadband MIMO Approaches

- OFDM (if approximate channel length is known):
  1. divide spectrum into narrowband channels;
  2. address each narrowband channel independently using narrowband-optimal techniques;

  drawback: ignores spectral coherence across frequency bins;
- optimum filter bank transceiver (if channel itself is known):
  1. block processing;
  2. inter-block interference is eliminated by guard intervals;
  3. resulting matrix can be diagonalised by SVD;
- both techniques invest DOFs into the guard intervals, which are generally not balanced against other error sources.



Polynomial Singular Value Decompositions

- Iterative algorithms have been developed to determine a polynomial eigenvalue decomposition (EVD) for a parahermitian matrix $\boldsymbol{R}(z) = \tilde{\boldsymbol{R}}(z) = \boldsymbol{R}^{\mathrm{H}}(z^{-1})$:

$$\boldsymbol{R}(z) \approx \boldsymbol{H}(z)\boldsymbol{\Gamma}(z)\tilde{\boldsymbol{H}}(z)$$

- paraunitary $\boldsymbol{H}(z)\tilde{\boldsymbol{H}}(z) = \mathbf{I}$, diagonal and spectrally majorised $\boldsymbol{\Gamma}(z)$;
- polynomial SVD of channel $\boldsymbol{C}(z)$ can be obtained via two EVDs:

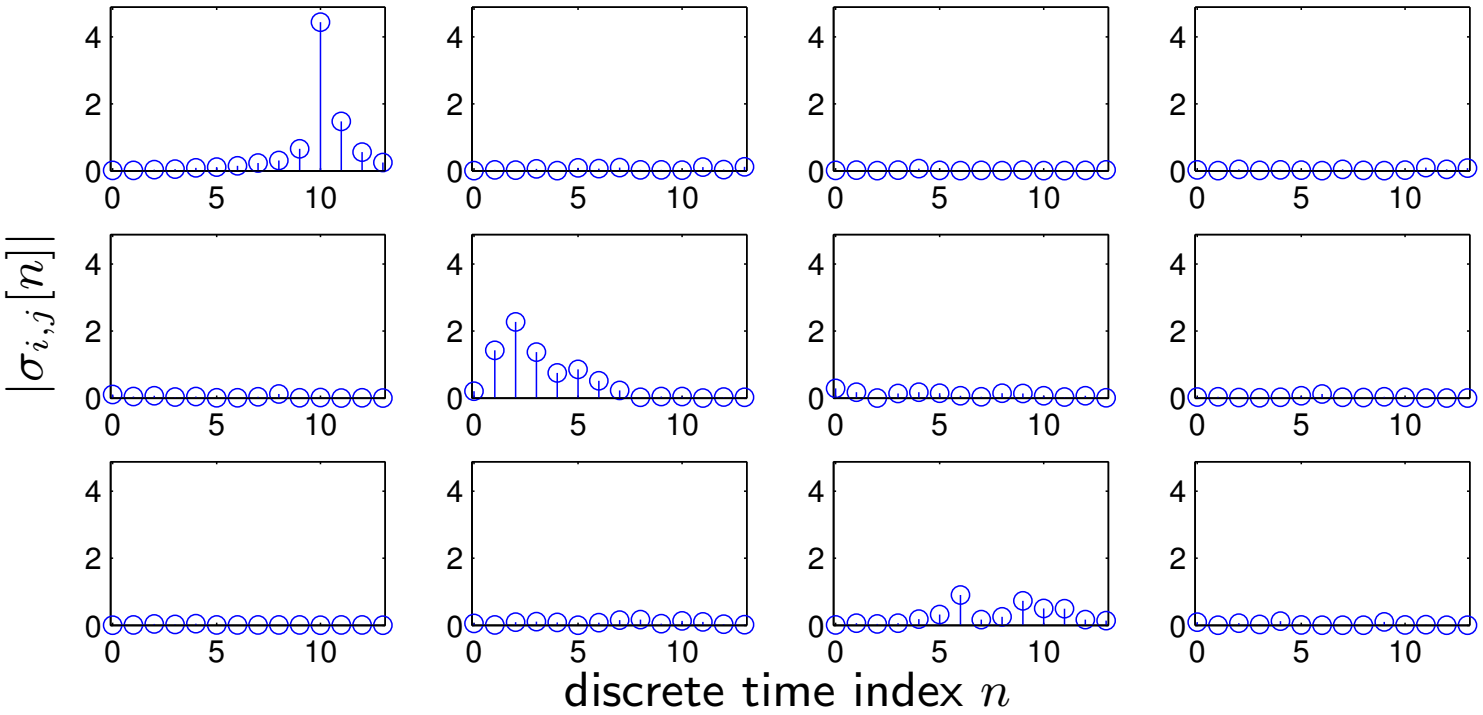
$$\boldsymbol{C}(z)\tilde{\boldsymbol{C}}(z) = \boldsymbol{U}(z)\boldsymbol{\Sigma}^{+}(z)\boldsymbol{\Sigma}^{-}(z)\tilde{\boldsymbol{U}}(z)$$
$$\tilde{\boldsymbol{C}}(z)\boldsymbol{C}(z) = \boldsymbol{V}(z)\boldsymbol{\Sigma}^{-}(z)\boldsymbol{\Sigma}^{+}(z)\tilde{\boldsymbol{V}}(z)$$

finally:

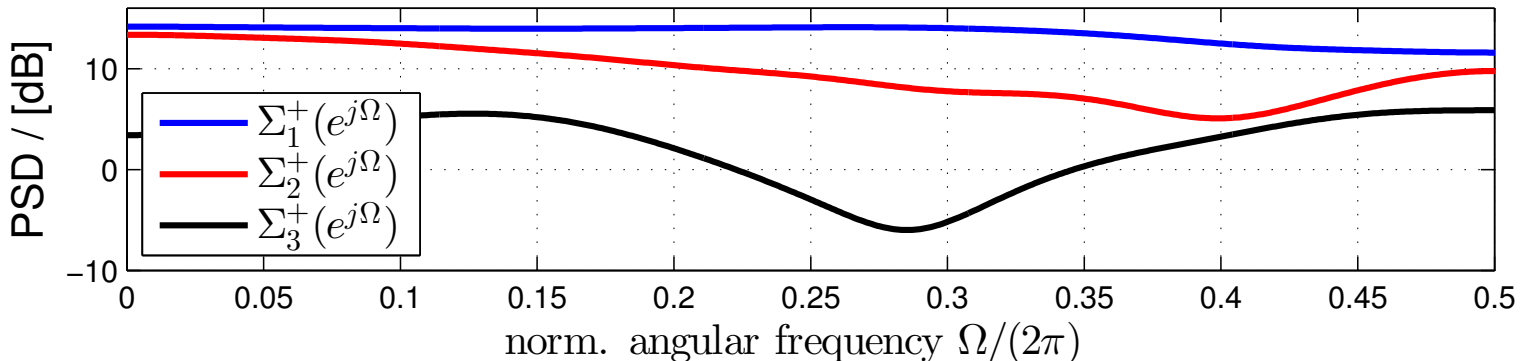
$$\boldsymbol{C}(z) = \boldsymbol{U}(z)\boldsymbol{\Sigma}^{+}(z)\tilde{\boldsymbol{V}}(z)$$

MIMO Application Example

- Polynomial SVD of the previous $\boldsymbol{C}(z) \in \mathbb{C}^{3\times 4}$ channel matrix:

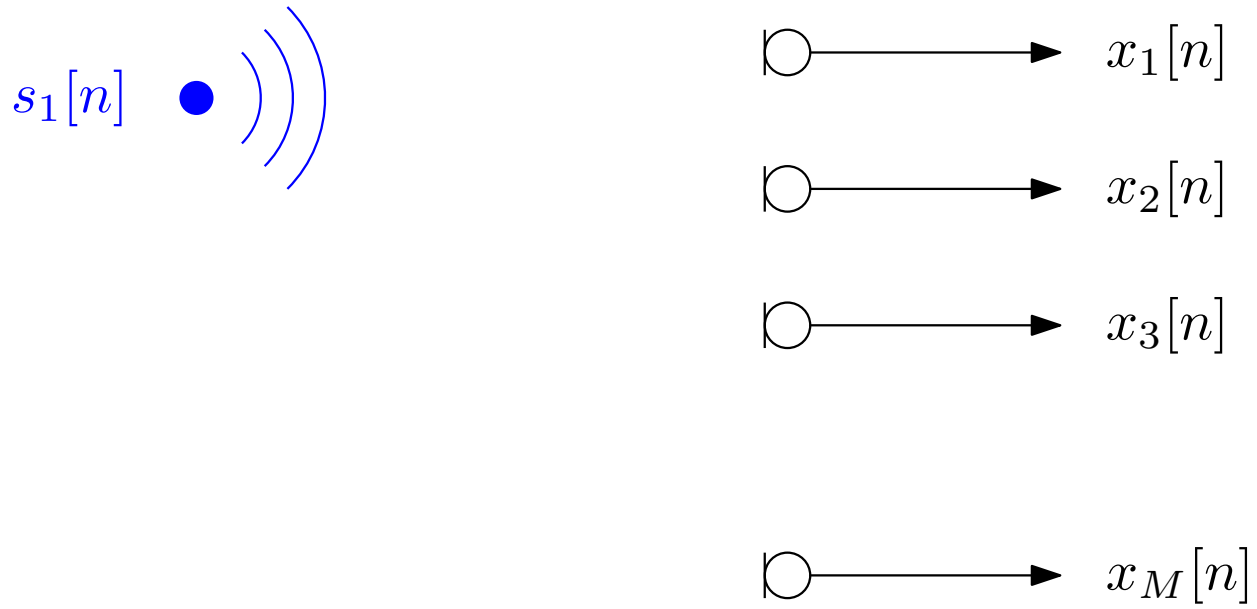


- the singular value spectra are majorised:



Narrowband Source Model

- Scenario with sensor array and far-field sources:

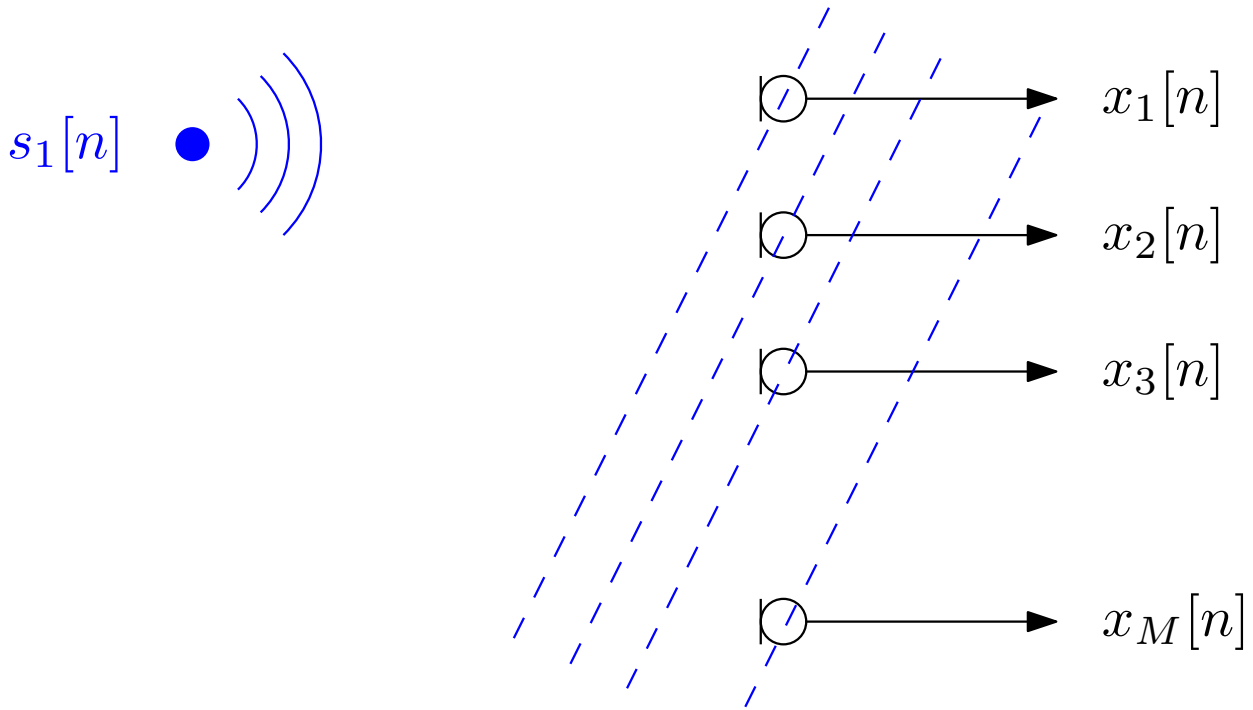


- for the narrowband case, the source signals arrive with delays, expressed by phase shifts in a steering vector
- data model:

$$\mathbf{x}[n] =$$

Narrowband Source Model

- Scenario with sensor array and far-field sources:

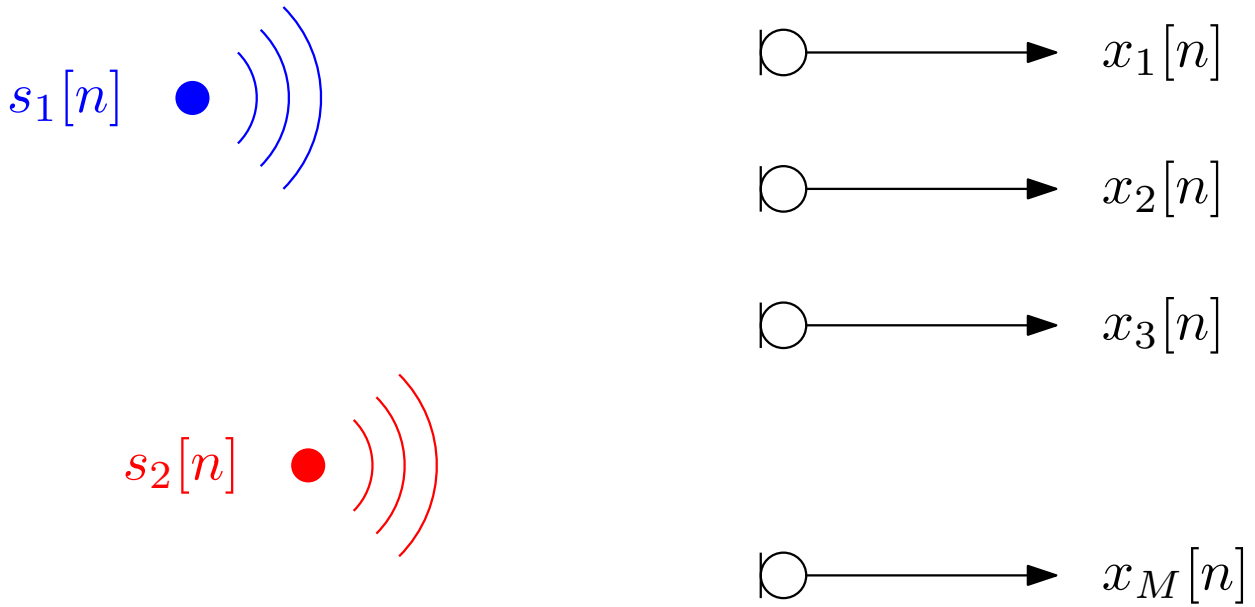


- for the narrowband case, the source signals arrive with delays, expressed by phase shifts in a steering vector $\mathbf{s}_1$
- data model:

$$\mathbf{x}[n] = s_1[n] \cdot \mathbf{s}_1$$

Narrowband Source Model

- Scenario with sensor array and far-field sources:

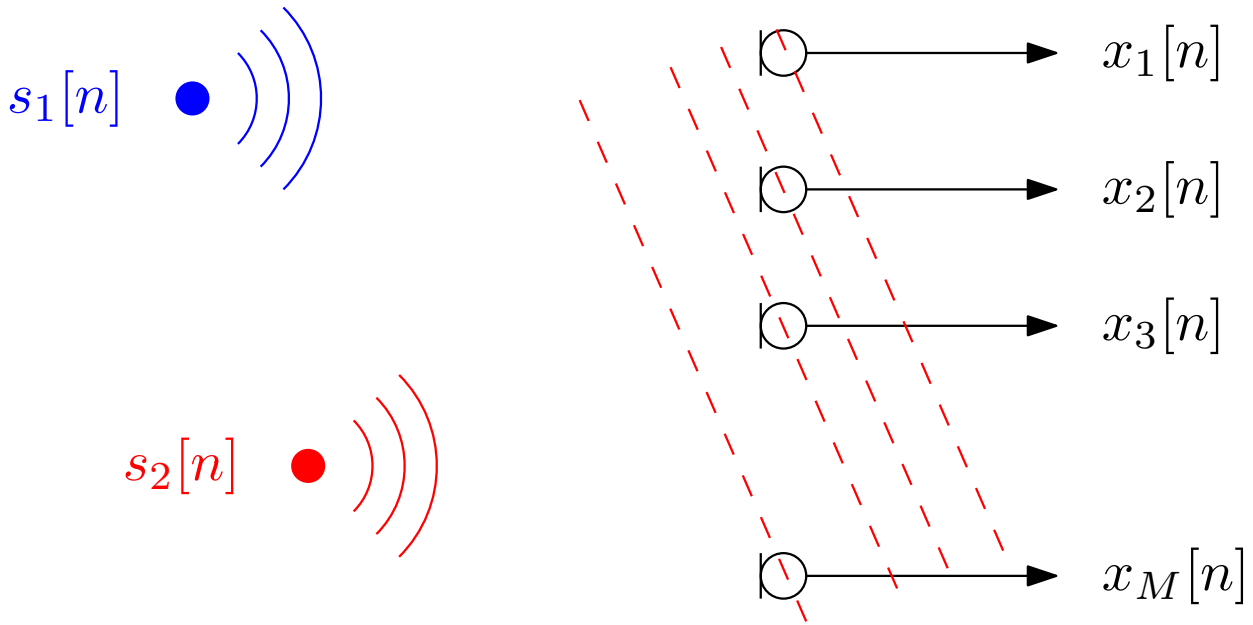


- for the narrowband case, the source signals arrive with delays, expressed by phase shifts in a steering vector $\mathbf{s}_1$
- data model:

$$\mathbf{x}[n] = s_1[n] \cdot \mathbf{s}_1$$

Narrowband Source Model

- Scenario with sensor array and far-field sources:

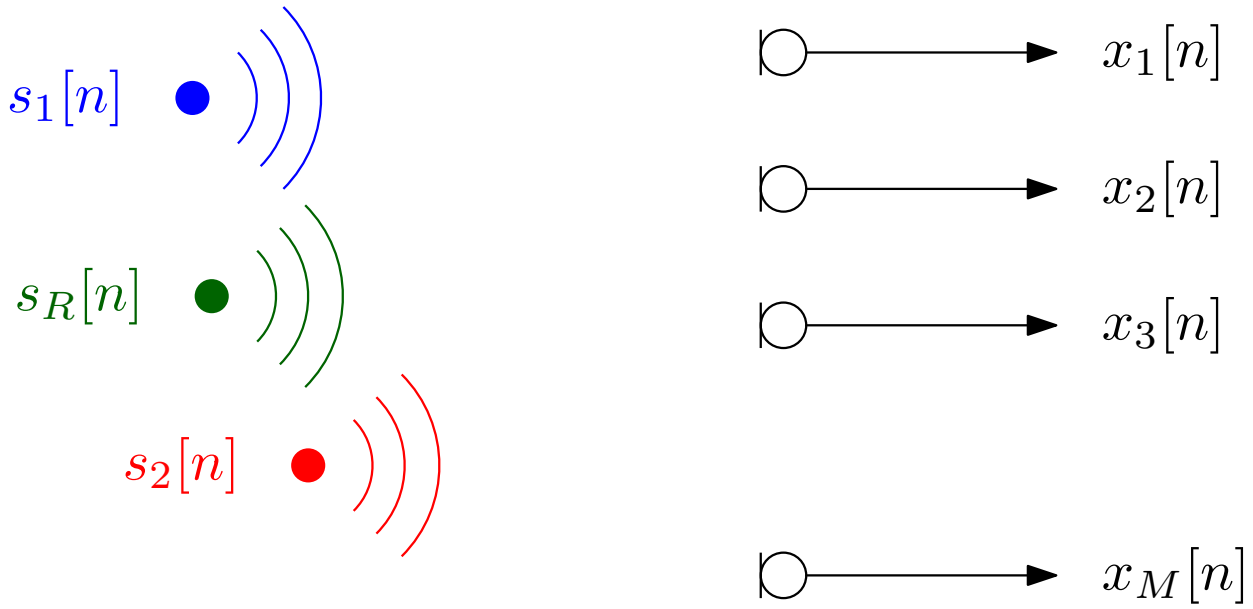


- for the narrowband case, the source signals arrive with delays, expressed by phase shifts in a steering vector $\mathbf{s}_1$, $\mathbf{s}_2$
- data model:

$$\mathbf{x}[n] = s_1[n] \cdot \mathbf{s}_1 + s_1[n] \cdot \mathbf{s_2}$$

Narrowband Source Model

- Scenario with sensor array and far-field sources:

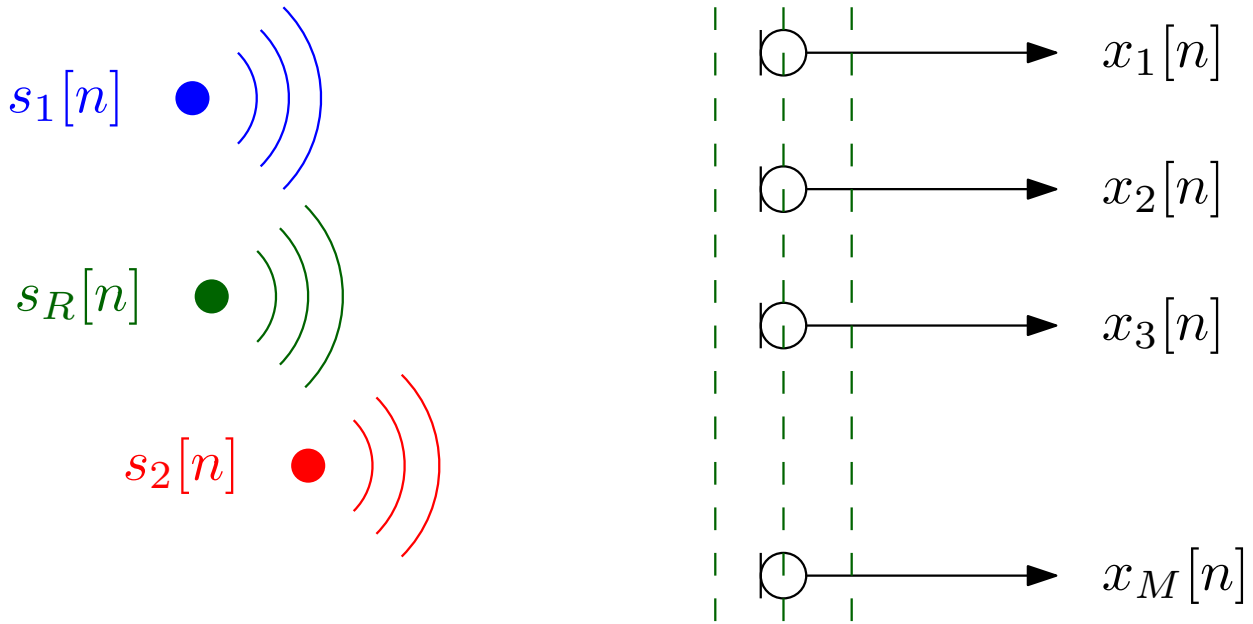


- for the narrowband case, the source signals arrive with delays, expressed by phase shifts in a steering vector $\mathbf{s}_1$, $\mathbf{s}_2$
- data model:

$$\mathbf{x}[n] = s_1[n] \cdot \mathbf{s}_1 + s_1[n] \cdot \mathbf{s_2}$$

Narrowband Source Model

- Scenario with sensor array and far-field sources:



- for the narrowband case, the source signals arrive with delays, expressed by phase shifts in a steering vector $\mathbf{s}_1$, $\mathbf{s}_2$, $\ldots \mathbf{s}_R$;
- data model:

$$\mathbf{x}[n] = s_1[n] \cdot \mathbf{s}_1 + s_1[n] \cdot \mathbf{s_2} + \cdots + s_R[n] \cdot \mathbf{s}_R = \sum_{r=1}^{R} s_r[n] \cdot \mathbf{s}_r$$

Steering Vector

- A signal $s[n]$ arriving at the array can be characterised by the delays of its wavefront (neglecting attenuation):

$$\begin{bmatrix} x_0[n] \\ x_1[n] \\ \vdots \\ x_{M-1}[n] \end{bmatrix} = \begin{bmatrix} s[n-\tau_0] \\ s[n-\tau_1] \\ \vdots \\ s[n-\tau_{M-1}] \end{bmatrix} = \begin{bmatrix} \delta[n-\tau_0] \\ \delta[n-\tau_1] \\ \vdots \\ \delta[n-\tau_{M-1}] \end{bmatrix} * s[n] \circ\!\!-\!\!\bullet\, \mathbf{a}_\vartheta(z) S(z)$$

- if evaluated at a narrowband normalised angular frequency Ω_i, the time delays τ_m in the broadband steering vector $\mathbf{a}_\vartheta(z)$ collapse to phase shifts in the narrowband steering vector $\mathbf{a}_{\vartheta,\Omega_i}$,

$$\mathbf{a}_{\vartheta,\Omega_i} = \mathbf{a}_\vartheta(z)\big|_{z=e^{j\Omega_i}} = \begin{bmatrix} e^{-j\tau_0\Omega_i} \\ e^{-j\tau_1\Omega_i} \\ \vdots \\ e^{-j\tau_{M-1}\Omega_i} \end{bmatrix} .$$

Data and Covariance Matrices

- A data matrix $\mathbf{X} \in \mathbb{C}^{M \times L}$ can be formed from L measurements:

$$\mathbf{X} = \begin{bmatrix} \mathbf{x}[n] & \mathbf{x}[n+1] & \dots & \mathbf{x}[n+L-1] \end{bmatrix}$$

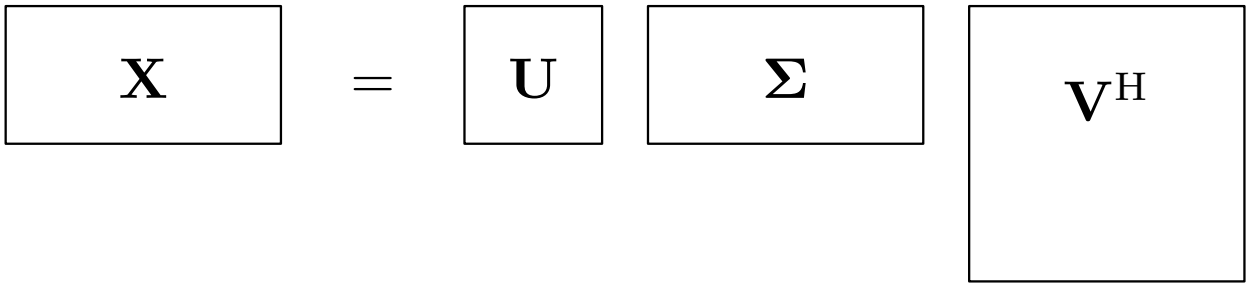
- assuming that all $x_m[n]$, $m = 1, 2, \dots M$ are zero mean, the (instantaneous) data covariance matrix is

$$\mathbf{R} = \mathcal{E}\{\mathbf{x}[n]\mathbf{x}^{\mathrm{H}}[n]\} \approx \frac{1}{L}\mathbf{X}\mathbf{X}^{\mathrm{H}}$$

  where the approximation assumes ergodicity and a sufficiently large L;
- Problem: can we tell from $\mathbf{X}$ or $\mathbf{R}$ (i) the number of sources and (ii) their orgin / time series?
- w.r.t. Jonathon Chamber's introduction, we here only consider the underdetermined case of more sensors than sources, $M \geq K$, and generally $L \gg M$.

SVD of Data Matrix

- Singular value decomposition of $\mathbf{X}$:



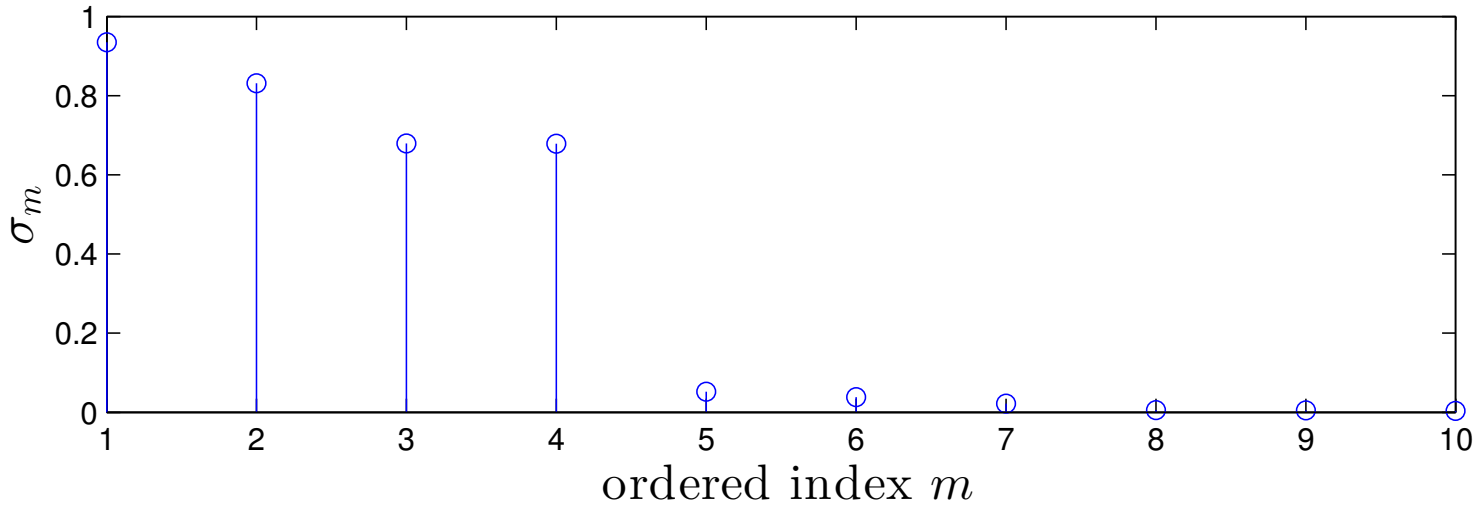
- unitary matrices $\mathbf{U} = [\mathbf{u}_1 \dots \mathbf{u}_M]$ and $\mathbf{V} = [\mathbf{v}_1 \dots \mathbf{v}_L]$;
- diagonal $\mathbf{\Sigma}$ contains the real, positive semidefinite singular values of $\mathbf{X}$ in descending order:

$$\mathbf{\Sigma} = \begin{bmatrix} \sigma_1 & 0 & \dots & 0 & 0 & \dots & 0 \\ 0 & \sigma_2 & \ddots & \vdots & \vdots & & \vdots \\ \vdots & \ddots & \ddots & 0 & \vdots & & \vdots \\ 0 & & 0 & \sigma_M & 0 & \dots & 0 \end{bmatrix}$$

with $\sigma_1 \geq \sigma_2 \geq \dots \geq \sigma_M \geq 0$.

Singular Values

- If the array is illuminated by $R \leq M$ linearly independent sources, the rank of
  the data matrix is
$$\text{rank}\{\mathbf{X}\} = R$$
- only the first R singular values of $\mathbf{X}$ will be non-zero;
- in practice, noise often will ensure that $\text{rank}\{\mathbf{X}\} = M$, with $M - R$ trailing singular values that define the noise floor:



- therefore, by thresholding singular values, it is possible to estimate the number of linearly independent sources R.

Subspace Decomposition

- If rank$\{\mathbf{X}\} = R$, the SVD can be split:

$$\mathbf{X} = [\mathbf{U}_s \;\; \mathbf{U}_n] \begin{bmatrix} \mathbf{\Sigma}_s & \mathbf{0} \\ \mathbf{0} & \mathbf{\Sigma}_n \end{bmatrix} \begin{bmatrix} \mathbf{V}_s^{\mathrm{H}} \\ \mathbf{V}_n^{\mathrm{H}} \end{bmatrix}$$

- with $\mathbf{U}_s \in \mathbb{C}^{M \times R}$ and $\mathbf{V}_s^{\mathrm{H}} \in \mathbb{C}^{R \times L}$ corresponding to the R largest singular values;
- $\mathbf{U}_s$ and $\mathbf{V}_s^{\mathrm{H}}$ define the signal-plus-noise subspace of $\mathbf{X}$:

$$\mathbf{X} = \sum_{m=1}^{M} \sigma_m \mathbf{u}_m \mathbf{v}_m^{\mathrm{H}} \approx \sum_{m=1}^{R} \sigma_m \mathbf{u}_m \mathbf{v}_m^{\mathrm{H}}$$

- the complements $\mathbf{U}_n$ and $\mathbf{V}_n^{\mathrm{H}}$,

$$\mathbf{U}_s^{\mathrm{H}} \mathbf{U}_n = \mathbf{0} \quad , \quad \mathbf{V}_s \mathbf{V}_n^{\mathrm{H}} = \mathbf{0}$$

define the noise-only subspace of $\mathbf{X}$.

SVD via Two EVDs

- Any Hermitian matrix $\mathbf{A} = \mathbf{A}^{\mathrm{H}}$ allows an eigenvalue decomposition

$$\mathbf{A} = \mathbf{Q}\mathbf{\Lambda}\mathbf{Q}^{\mathrm{H}}$$

with $\mathbf{Q}$ unitary and the eigenvalues in $\mathbf{\Lambda}$ real valued and positive semi-definite;

- postulating $\mathbf{X} = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^{\mathrm{H}}$, therefore:

$$\mathbf{X}\mathbf{X}^{\mathrm{H}} = (\mathbf{U}\mathbf{\Sigma}\mathbf{V}^{\mathrm{H}})(\mathbf{V}\mathbf{\Sigma}^{\mathrm{H}}\mathbf{U}^{\mathrm{H}}) = \mathbf{U}\mathbf{\Lambda}\mathbf{U}^{\mathrm{H}} \quad (20)$$

$$\mathbf{X}^{\mathrm{H}}\mathbf{X} = (\mathbf{V}\mathbf{\Sigma}^{\mathrm{H}}\mathbf{U}^{\mathrm{H}})(\mathbf{U}\mathbf{\Sigma}\mathbf{V}^{\mathrm{H}}) = \mathbf{V}\mathbf{\Lambda}\mathbf{V}^{\mathrm{H}} \quad (21)$$

- (ordered) eigenvalues relate to the singular values: $\lambda_m = \sigma_m^2$;
- the covariance matrix $\mathbf{R} = \frac{1}{L}\mathbf{X}\mathbf{X}$ has the same rank as the data matrix $\mathbf{X}$, and with $\mathbf{U}$ provides access to the same spatial subspace decomposition.

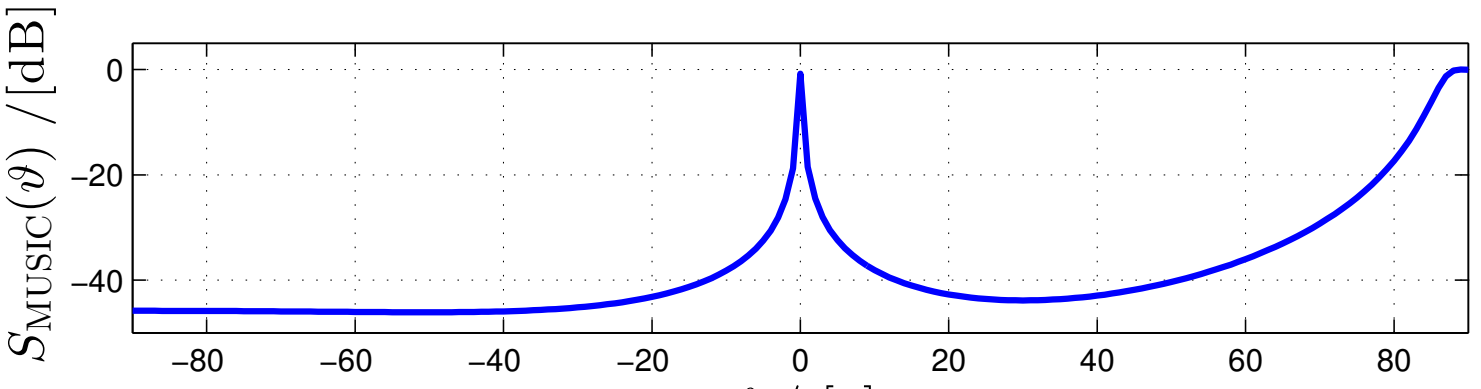
Narrowband MUSIC Algorithm

- EVD of the narrowband covariance matrix identifies signal-plus-noise and noise-only subspaces

$$\mathbf{R} = [\mathbf{U}_s \;\; \mathbf{U}_n] \begin{bmatrix} \mathbf{\Lambda}_s & \mathbf{0} \\ \mathbf{0} & \mathbf{\Lambda}_n \end{bmatrix} \begin{bmatrix} \mathbf{U}_s^{\mathrm{H}} \\ \mathbf{U}_n^{\mathrm{H}} \end{bmatrix}$$

- scanning the signal-plus-noise subspace could only help to retrieve sources with orthogonal steering vectors;
- therefore, the multiple signal classification (MUSIC) algorithm scans the noise-only subspace for minima, or maxima of its reciprocal

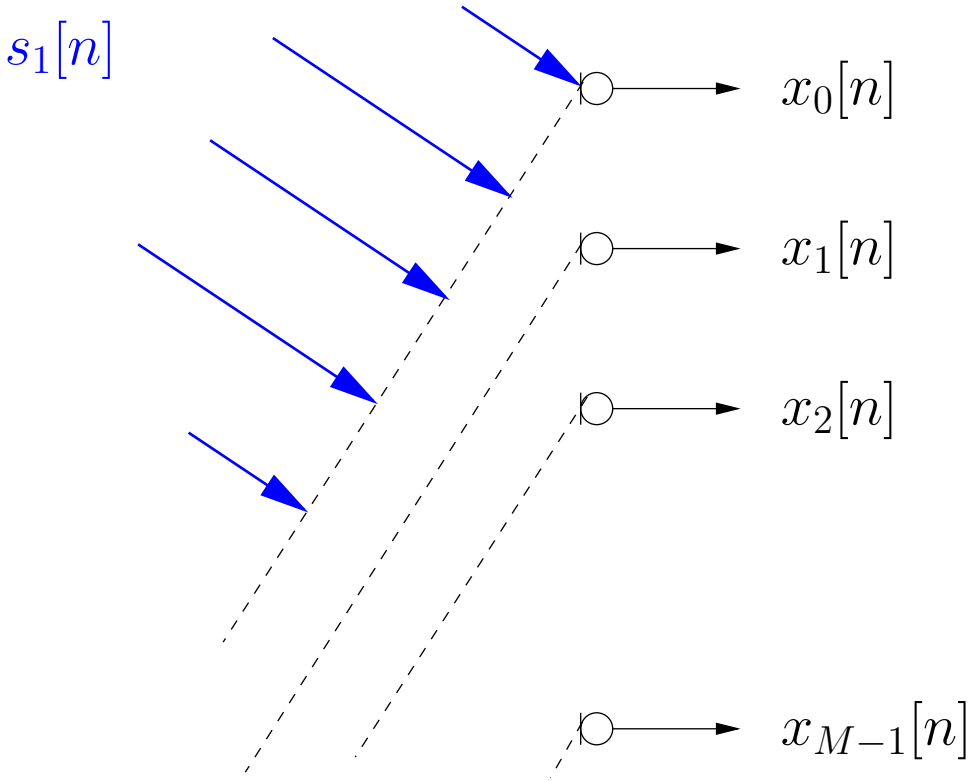
$$S_{\mathrm{MUSIC}}(\vartheta) = \frac{1}{\|\mathbf{U}_n \mathbf{a}_{\vartheta,\Omega_i}\|_2^2}$$



Narrowband Source Separation

- Via SVD of the data matrix $\mathbf{X}$ or EVD of the covariance matrix $\mathbf{R}$, we can determine the number of linearly independent sources R;
- using the subspace decompositions offered by EVD/SVD, the directions of arrival can be estimated using e.g. MUSIC;
- based on knowledge of the angle of arrival, beamforming could be applied to $\mathbf{X}$ to extract specific sources;
- overall: EVD (and SVD) can play a vital part in narrowband source separation;
- what about broadband source separation?

Broadband Array Scenario



- Compared to the narrowband case, time delays rather than phase shifts bear information on the direction of a source.

Broadband Steering Vector

- A signal $s[n]$ arriving at the array can be characterised by the delays of its wavefront (neglecting attenuation):

$$\begin{bmatrix} x_0[n] \\ x_1[n] \\ \vdots \\ x_{M-1}[n] \end{bmatrix} = \begin{bmatrix} s[n-\tau_0] \\ s[n-\tau_1] \\ \vdots \\ s[n-\tau_{M-1}] \end{bmatrix} = \begin{bmatrix} \delta[n-\tau_0] \\ \delta[n-\tau_1] \\ \vdots \\ \delta[n-\tau_{M-1}] \end{bmatrix} * s[n] \circ\!\!-\!\!\bullet\, \mathbf{a}_\vartheta(z) S(z)$$

- if evaluated at a narrowband normalised angular frequency Ω_i, the time delays τ_m in the broadband steering vector $\mathbf{a}_\vartheta(z)$ collapse to phase shifts in the narrowband steering vector $\mathbf{a}_{\vartheta,\Omega_i}$,

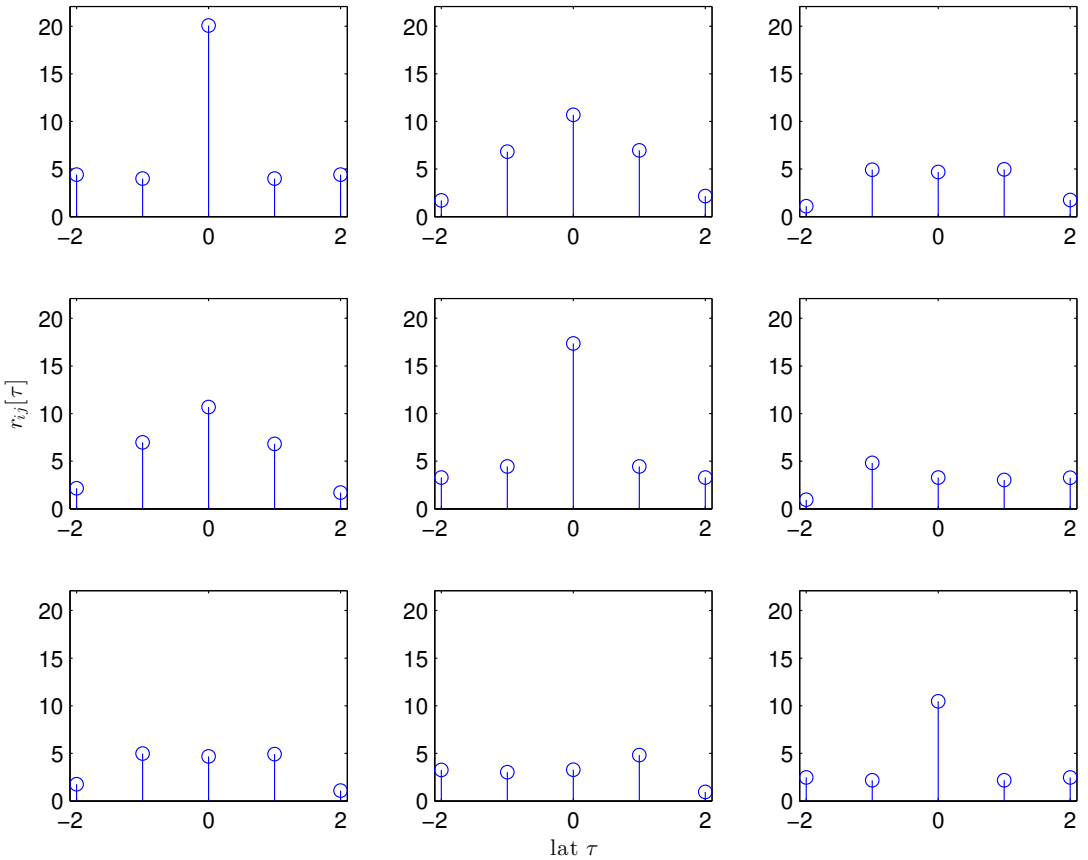
$$\mathbf{a}_{\vartheta,\Omega_i} = \mathbf{a}_\vartheta(z)\big|_{z=e^{j\Omega_i}} = \begin{bmatrix} e^{-j\tau_0\Omega_i} \\ e^{-j\tau_1\Omega_i} \\ \vdots \\ e^{-j\tau_{M-1}\Omega_i} \end{bmatrix} .$$

Space-Time Covariance Matrix

- If delays must be considered, the (space-time) covariance matrix must capture the lag τ:

$$\mathbf{R}[\tau] = \mathcal{E}\{\mathbf{x}[n] \cdot \mathbf{x}^{\mathrm{H}}[n-\tau]\}$$

- $\mathbf{R}[\tau]$ contains auto- and cross-correlation sequences:



Cross Spectral Density Matrix

- z-transform of the space-time covariance matrix is given by

$$\mathbf{R}[\tau] = \mathcal{E}\{\mathbf{x}_n \mathbf{x}_{n-\tau}^{\mathrm{H}}\} \quad \circ\!\!-\!\!\bullet \quad \boldsymbol{R}(z) = \sum_l S_l(z)\mathbf{a}_{\vartheta_l}(z)\tilde{\mathbf{a}}_{\vartheta_l}(z) + \sigma_N^2 \mathbf{I}$$

  with ϑ_l the direction of arrival and $S_l(z)$ the PSD of the lth source;
- $\boldsymbol{R}(z)$ is the cross spectral density (CSD) matrix;
- the instantaneous covariance matrix (no lag parameter τ)

$$\mathbf{R} = \mathcal{E}\{\mathbf{x}_n \mathbf{x}_n^{\mathrm{H}}\} = \mathbf{R}[0]$$

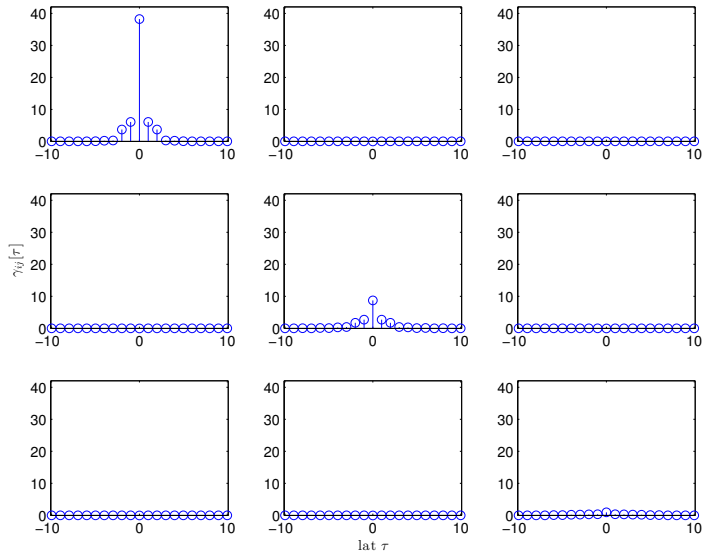
Polynomial MUSIC (PMUSIC)

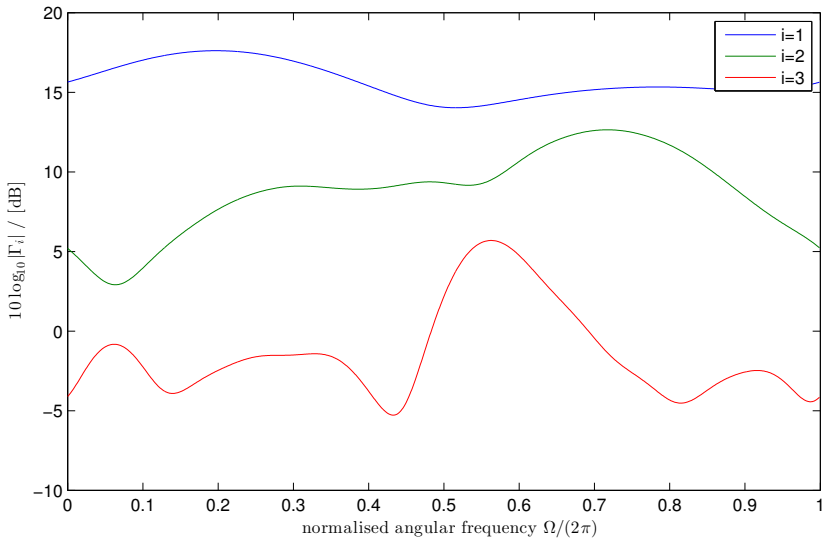
[Alrmah, Weiss, Lambotharan, *EUSIPCO* (2011)]

- Based on the polynomial EVD of the broadband covariance matrix

$$\mathbf{R}(z) \approx \underbrace{[\mathbf{Q}_s(z) \;\; \mathbf{Q}_n(z)]}_{\mathbf{Q}(z)} \underbrace{\begin{bmatrix} \mathbf{\Lambda}_s(z) & \mathbf{0} \\ \mathbf{0} & \mathbf{\Lambda}_n(z) \end{bmatrix}}_{\mathbf{\Lambda}(z)} \begin{bmatrix} \tilde{\mathbf{Q}}_s(z) \\ \tilde{\mathbf{Q}}_n(z) \end{bmatrix}$$

- paraunitary $\mathbf{Q}(z)$, s.t. $\mathbf{Q}(z)\tilde{\mathbf{Q}}(z) = \mathbf{I}$;
- diagonalised and spectrally majorised $\mathbf{\Lambda}(z)$:





PMUSIC cont'd

- Idea —- scan the polynomial noise-only subspace $Q_n(z)$ with broadband steering vectors

$$\Gamma(z,\vartheta) = \tilde{\mathbf{a}}_\vartheta(z)\tilde{\mathbf{Q}}_n(z)\mathbf{Q}_n(z)\mathbf{a}_\vartheta(z)$$

- looking for minima leads to a spatio-spectral PMUSIC

$$S_{\text{PSS-MUSIC}}(\vartheta,\Omega) = \left(\Gamma(z,\vartheta)|_{z=e^{j\Omega}}\right)^{-1}$$

- and a spatial-only PMUSIC

$$S_{\text{PS-MUSIC}}(\vartheta) = \left(2\pi \oint \Gamma(z,\vartheta)|_{z=e^{j\Omega}} d\Omega\right)^{-1} = \Gamma_\vartheta^{-1}[0]$$

with $\Gamma_\vartheta[\tau] \circ\!\!-\!\!\bullet\, \Gamma(z,\vartheta)$.

Simulation I — Toy Problem

- Linear uniform array with critical spatial and temporal sampling;
- broadband steering vector for end-fire position:

$$\mathbf{a}_{\pi/2}(z) = \begin{bmatrix} 1 & z^{-1} & \cdots & z^{-M+1} \end{bmatrix}^{\mathrm{T}}$$

- covariance matrix

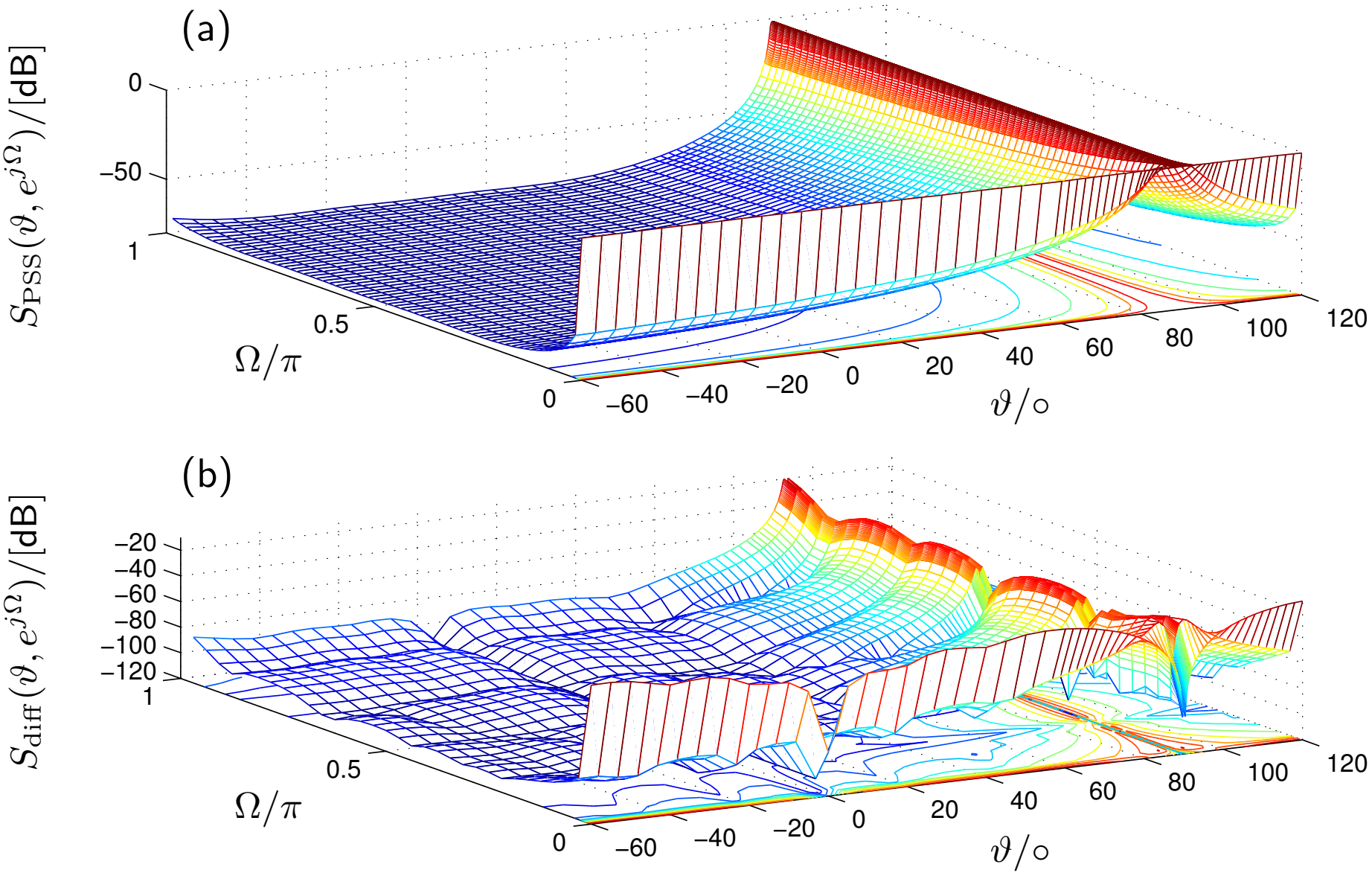
$$\mathbf{R}(z) = \mathbf{a}_{\pi/2}(z)\tilde{\mathbf{a}}_{\pi/2}(z) = \begin{bmatrix} 1 & z^{1} & \cdots & z^{M-1} \\ z^{-1} & 1 & & \vdots \\ \vdots & & \ddots & \vdots \\ z^{-M+1} & \cdots & \cdots & 1 \end{bmatrix} .$$

- PEVD (by inspection)

$$\mathbf{Q}(z) = \mathbf{T}_{\mathrm{DFT}}\mathrm{diag}\{1 \;\; z^{-1} \;\; \cdots \;\; z^{-M+1}\} \quad ; \qquad \mathbf{\Lambda}(z) = \mathrm{diag}\{1\; 0 \;\cdots\; 0$$

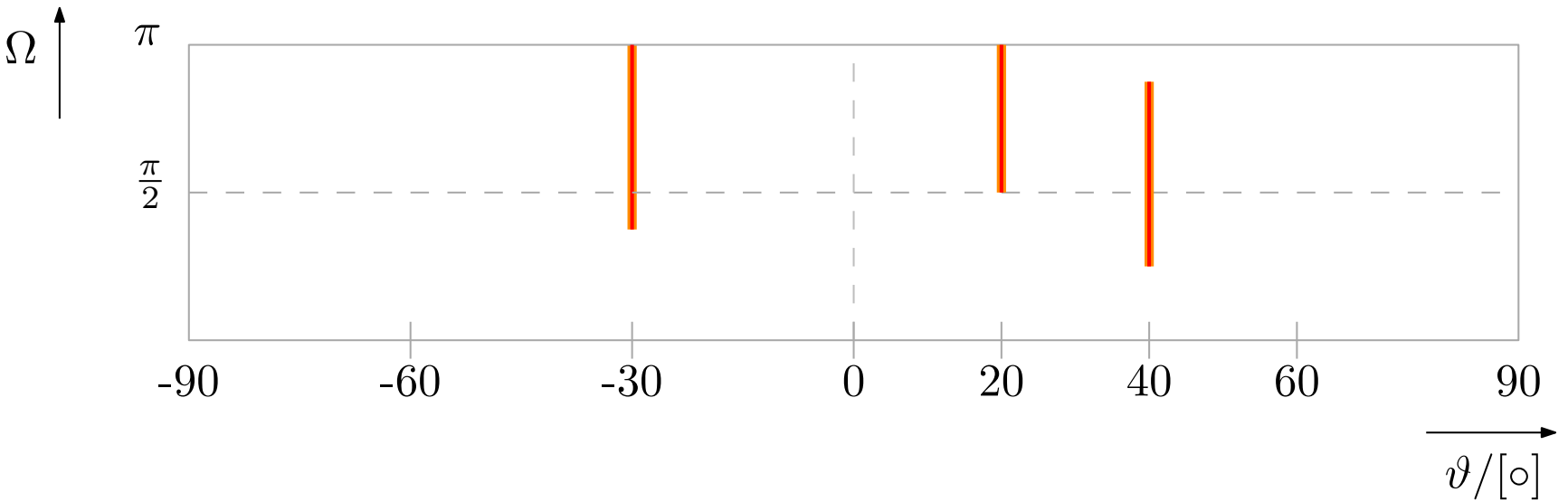
- simulations with $M = 4$ ...

Simulation I — PSS-MUSIC



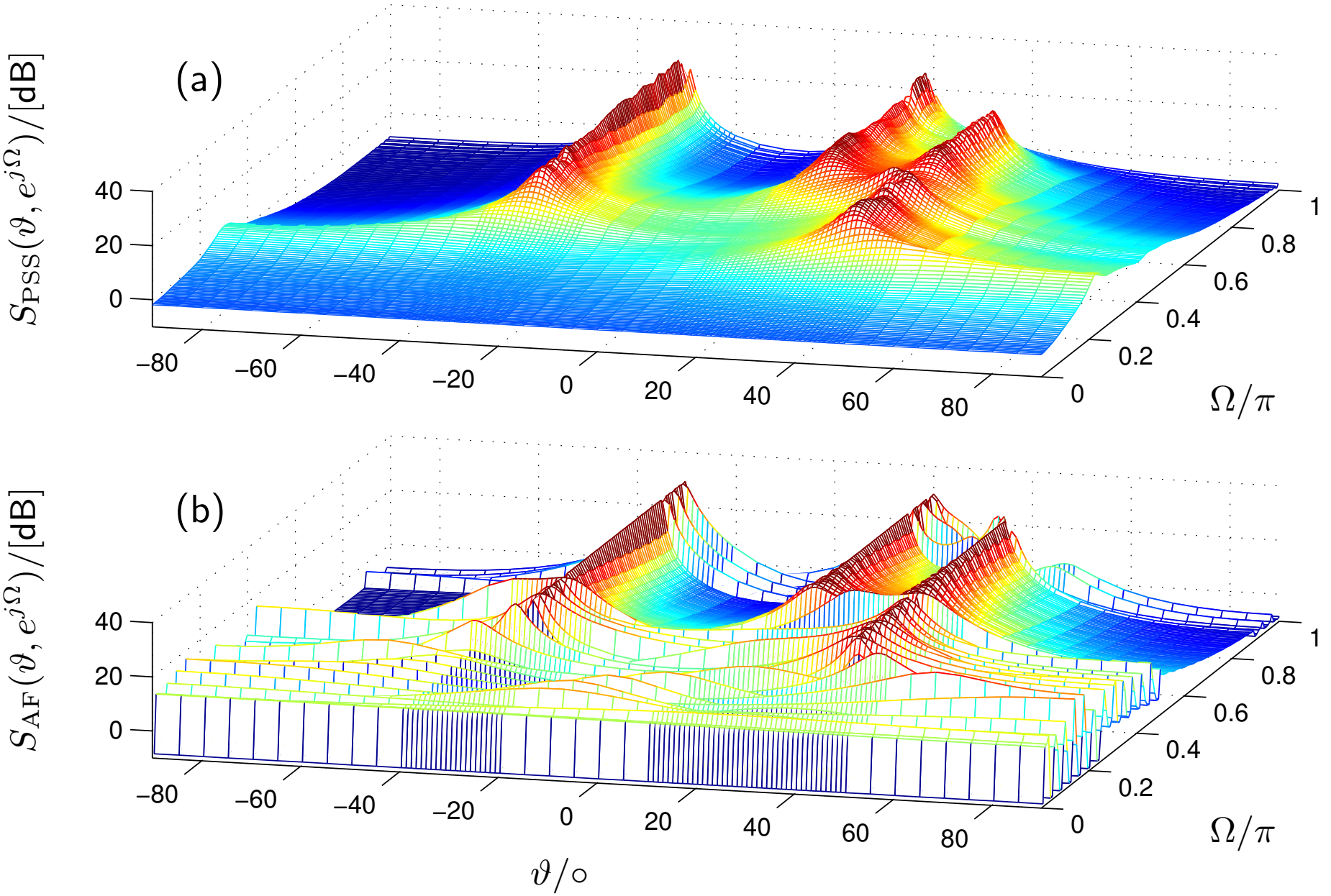
Simulation II

- $M = 8$ element sensor array illuminated by three sources;
- source 1: $\vartheta_1 = -30°$, active over range $\Omega \in [\frac{3\pi}{8};\ \pi]$;
- source 2: $\vartheta_2 = 20°$, active over range $\Omega \in [\frac{\pi}{2};\ \pi]$;
- source 3: $\vartheta_3 = 40°$, active over range $\Omega \in [\frac{2\pi}{8};\ \frac{7\pi}{8}]$; and



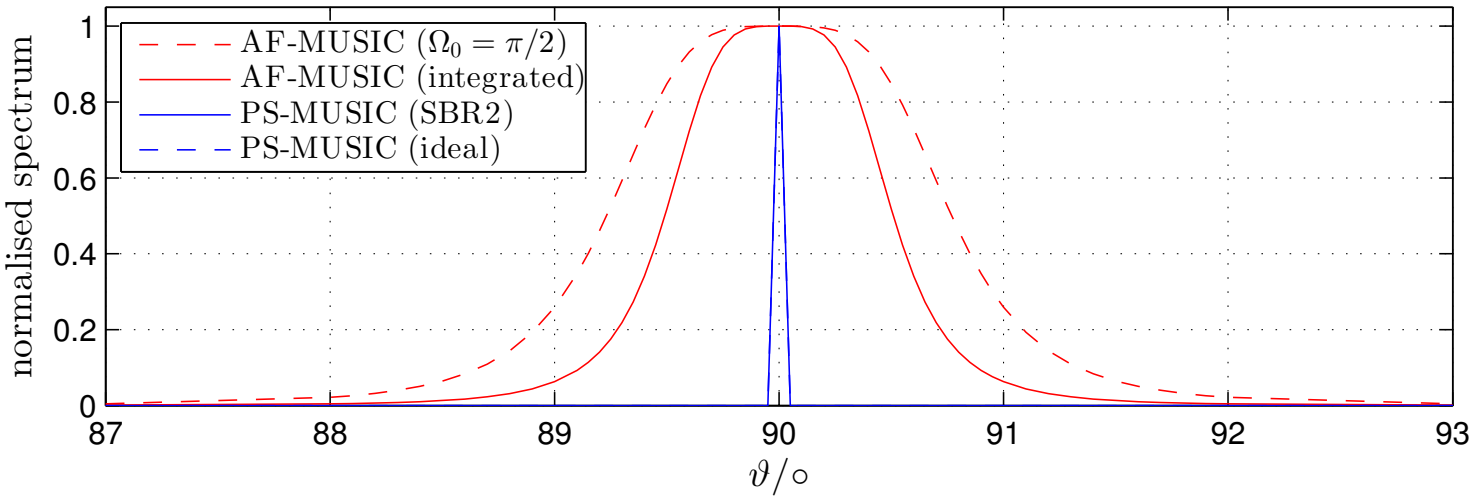
- filter banks as innovation filters, and broadband steering vectors to simulate AoA;
- space-time covariance matrix is estimated from 10^4 samples.

Simulation II — PSS-MUSIC

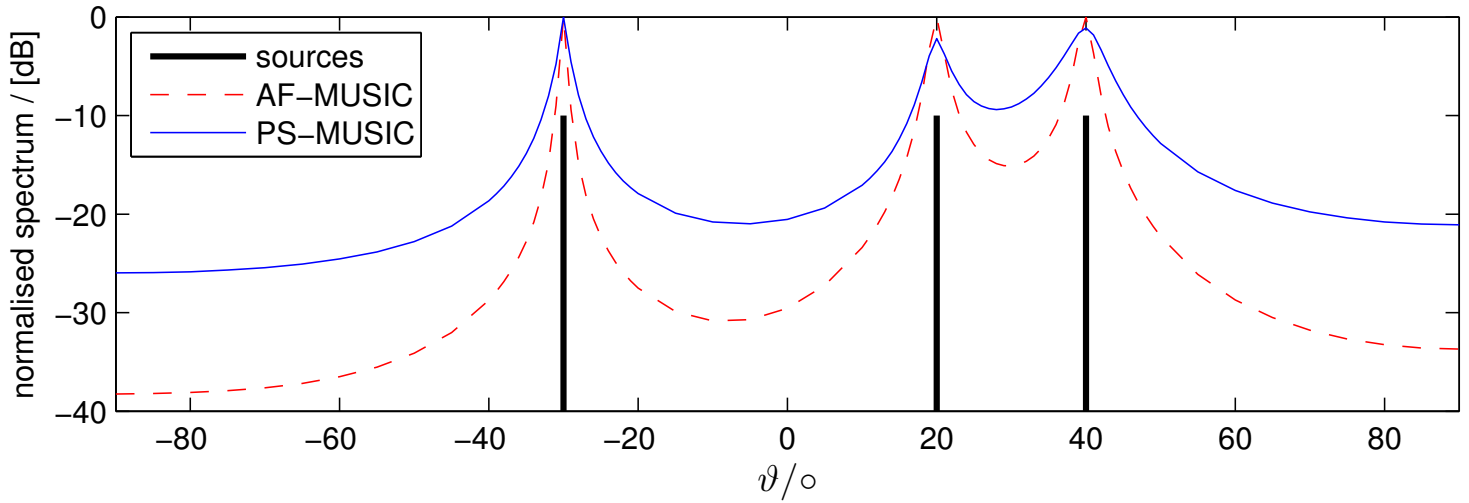


PS-MUSIC Comparison

- Simulation I (toy problem): peaks normalised to unity:



- Simulation II: inaccuracies on PEVD and broadband steering vector

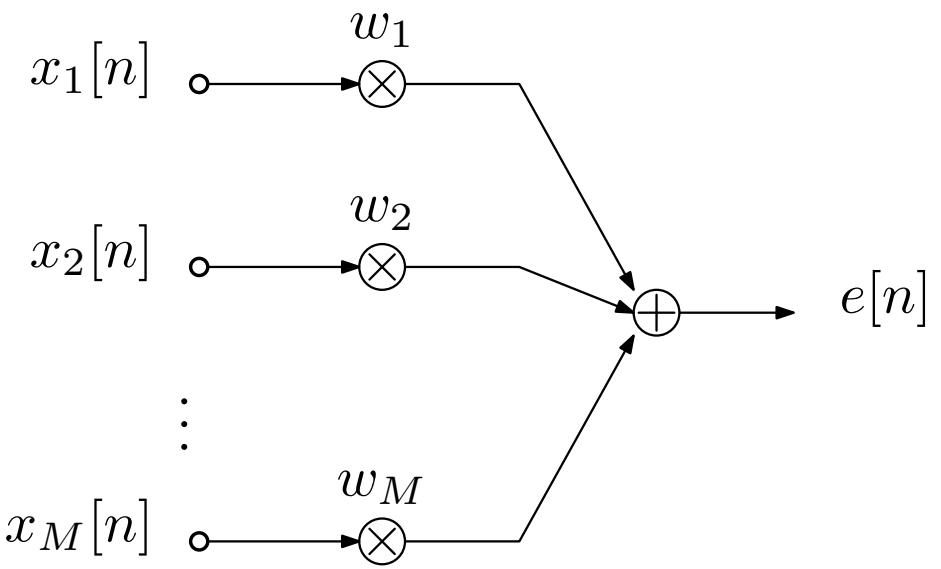


AoA Estimation — Conclusions

- We have considered the importance of SVD and EVD for narrowband source separation;
- narrowband matrix decomposition real the matrix rank and offer subspace decompositions on which angle-of-arrival estimation alhorithms such as MUSIC can be based;
- broadband problems lead to a space-time covariance or CSD matrix;
- such polynomial matrices cannot be decomposed by standard EVD and SVD;
- a polynomial EVD has been defined;
- iterative algorithms such as SBR2 can be used to approximate the PEVD;
- this permits a number of applications, such as broadband angle of arrival estimation;
- broadband beamforming could then be used to separate broadband sources.

Narrowband Minimum Variance Distortionless Response Beamformer

- Scenario: an array of M sensors receives data $\mathbf{x}[n]$, containing a desired signal with frequency Ω_{s} and angle of arrival ϑ_{s}, corrupted by interferers;
- a narrowband beamformer applies a single coefficient to every of the M sensor signals:



Narrowband MVDR Problem

- Recall the space-time covariance matrix:

$$\mathbf{R}[\tau] = \mathcal{E}\{\mathbf{x}[n]\mathbf{x}^{\mathrm{H}}[n-\tau]\}$$

- the MVDR beamformer minimises the output power of the beamformer:

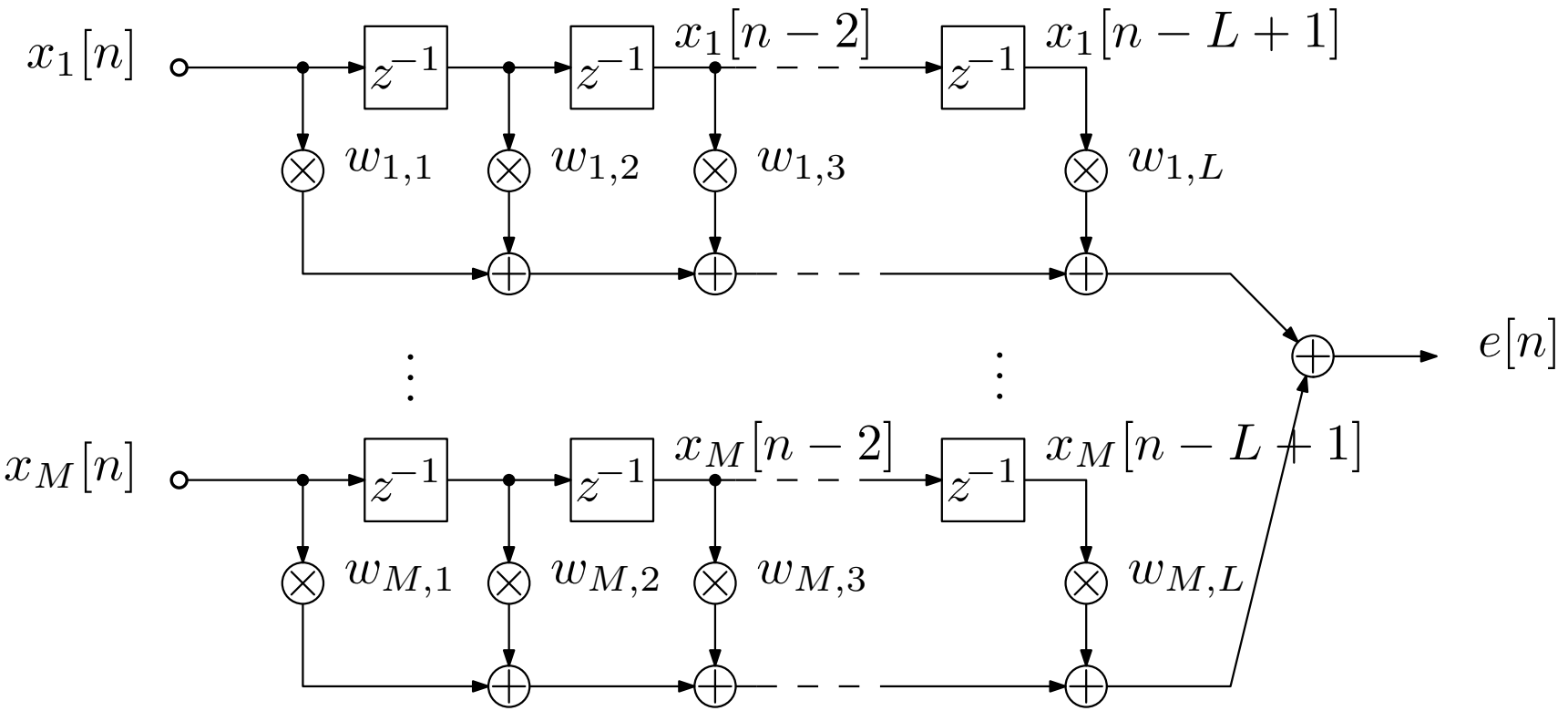
$$\min_{\mathbf{w}} \mathcal{E}\{|e[n]|^2\} = \min_{\mathbf{w}} \mathbf{w}^{\mathrm{H}}\mathbf{R}[0]\mathbf{w} \tag{22}$$

$$\text{s.t.} \quad \mathbf{a}^{\mathrm{H}}(\vartheta_{\mathrm{s}}, \Omega_{\mathrm{s}})\mathbf{w} = 1\ , \tag{23}$$

- this is subject to protecting the signal of interest by a constraint in look direction ϑ_{s};
- the steering vector $\mathbf{a}_{\vartheta_{\mathrm{s}},\Omega_{\mathrm{s}}}$ defines the signal of interest's parameters.

Broadband MVDR Beamformer

- Each sensor is followed by a tap delay line of dimension L, giving a total of ML coefficients in a vector $\mathbf{v} \in \mathbb{C}^{ML}$



Broadband MVDR Beamformer

- A larger input vector $\mathbf{x}_n \in \mathbb{C}^{ML}$ is generated, also including lags;
- the general approach is similar to the narrowband system, minimising the power of $e[n] = \mathbf{v}^{\mathrm{H}}\mathbf{x}_n$;
- however, we require several constraint equations to protect the signal of interest, e.g.

$$\mathbf{C} = [\mathbf{s}(\vartheta_{\mathrm{s}}, \Omega_0), \;\; \mathbf{s}(\vartheta_{\mathrm{s}}, \Omega_1) \;\; \ldots \;\; \mathbf{s}(\vartheta_{\mathrm{s}}, \Omega_{L-1})] \tag{24}$$

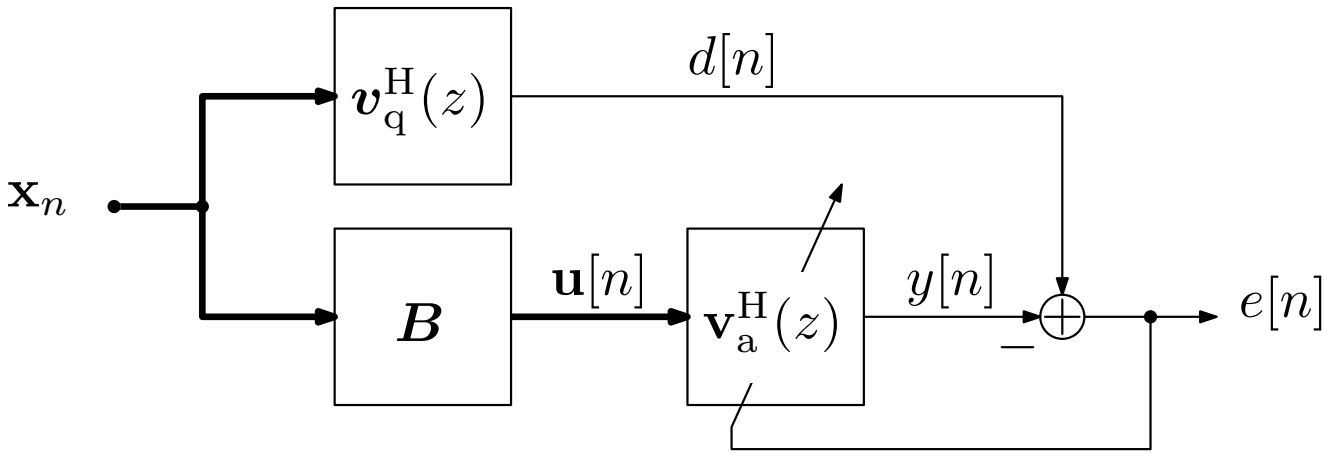
- these L constraints pin down the response to unit gain at L separate points in frequency:

$$\mathbf{C}^{\mathrm{H}}\mathbf{v} = \mathbf{1}\,; \tag{25}$$

- generally $\mathbf{C} \in \mathbb{C}^{ML \times L}$, but simplifications can be applied if the look direction is towards broadside.

Generalised Sidelobe Canceller

- A quiescent beamformer $\mathbf{v}_{\mathrm{q}} = \left(\mathbf{C}^{\mathrm{H}}\right)^{\dagger} \mathbf{1} \in \mathbb{C}^{ML}$ picks the signal of interest;
- the quiescent beamformer is optimal for AWGN but generally passes structured interference;
- the output of the blocking matrix $\mathbf{B}$ contains interference only, which requires $[\mathbf{BC}]$ to be unitary; hence $\mathbf{B} \in \mathbb{C}^{ML \times (M-1)L}$;
- an adaptive noise canceller $\mathbf{v}_{\mathrm{a}} \in \mathbb{C}^{(M-1)L}$ aims to remove the residual interference:



- note: all dimensions are determined by $\{M, L\}$.

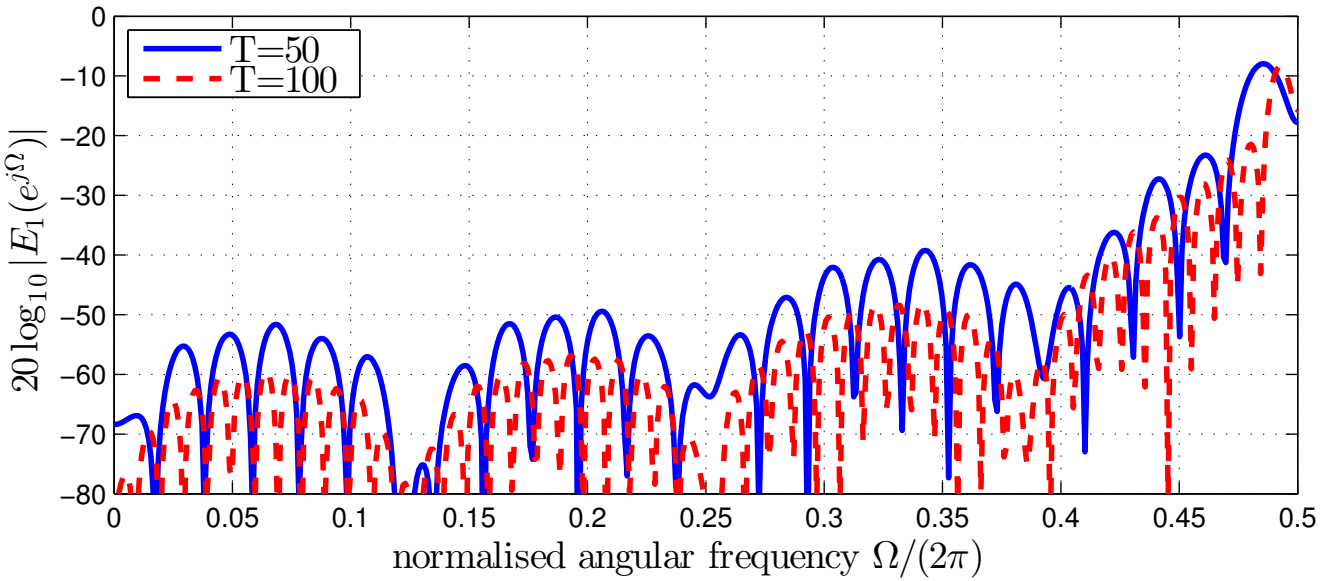
Polynomial Matrix MVDR Formulation

- Power spectral density of beamformer output: $R_e(z) = \tilde{\boldsymbol{w}}(z)\boldsymbol{R}(z)\boldsymbol{w}(z)$
- proposed broadband MVDR beamformer formulation:

$$\min_{\mathbf{w}(z)} \oint_{|z|=1} R_e(z)\frac{dz}{z} \tag{26}$$

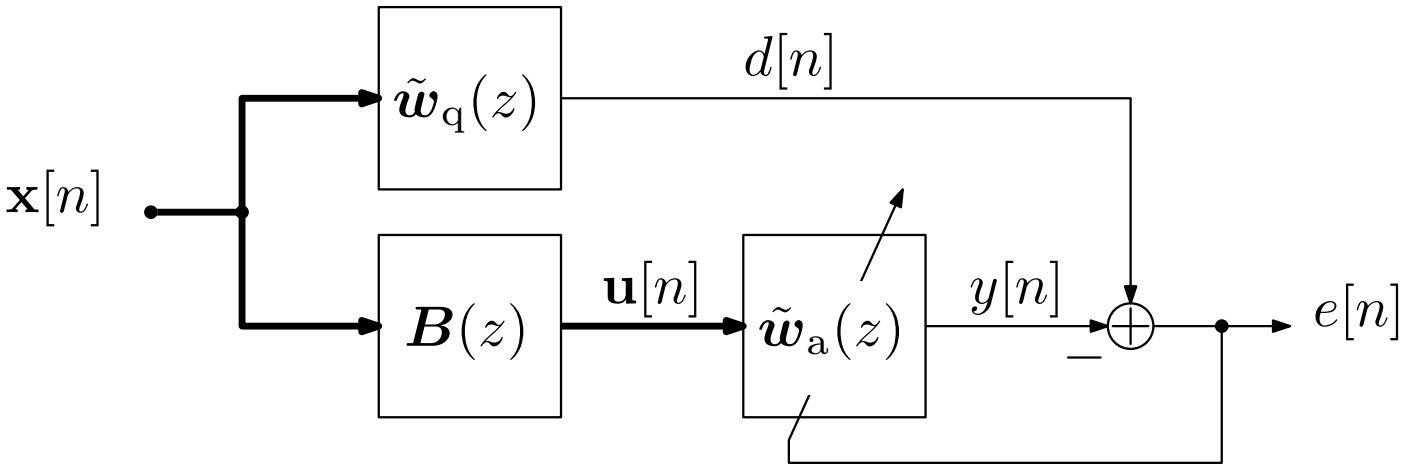
$$\text{s.t.} \quad \tilde{\boldsymbol{a}}(\vartheta_\mathrm{s}, z)\boldsymbol{w}(z) = F(z)\ . \tag{27}$$

- precision of broadband steering vector, $|\tilde{\boldsymbol{a}}(\vartheta_\mathrm{s}, z)\boldsymbol{a}(\vartheta_\mathrm{s}, z) - 1|$, depends on the length T of the fractional delay filter:



Generalised Sidelobe Canceller

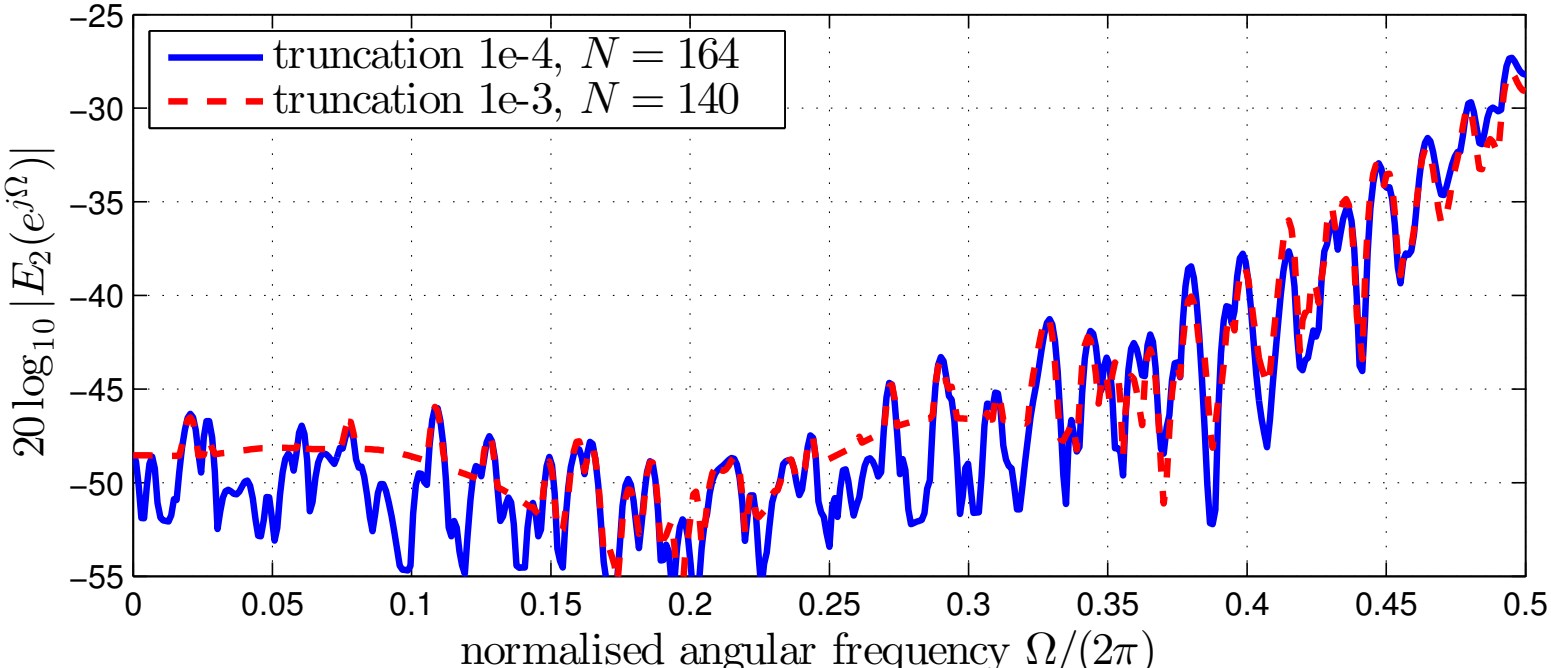
- Instead of performing constrained optimisation, the GSC projects the data and performs adaptive noise cancellation:



- the quiescent vector $\mathbf{w}_{\mathrm{q}}(z)$ is generated from the constraints and passes signal plus interference;
- the blocking matrix $\boldsymbol{B}(z)$ has to be orthonormal to $\mathbf{w}_{\mathrm{q}}(z)$ and only pass interference.

Design Considerations

- The blocking matrix can be obtained by completing a paraunitary matrix from $\mathbf{w}_{\mathrm{q}}(z)$;
- this can be achieved by calculating a PEVD of the rank one matrix $\mathbf{w}_{\mathrm{q}}(z)\tilde{\mathbf{w}}_{\mathrm{q}}(z)$;
- this leads to a block matrix of order N that is typically greater than L;
- maximum leakage of the signal of interest through the blocking matrix:



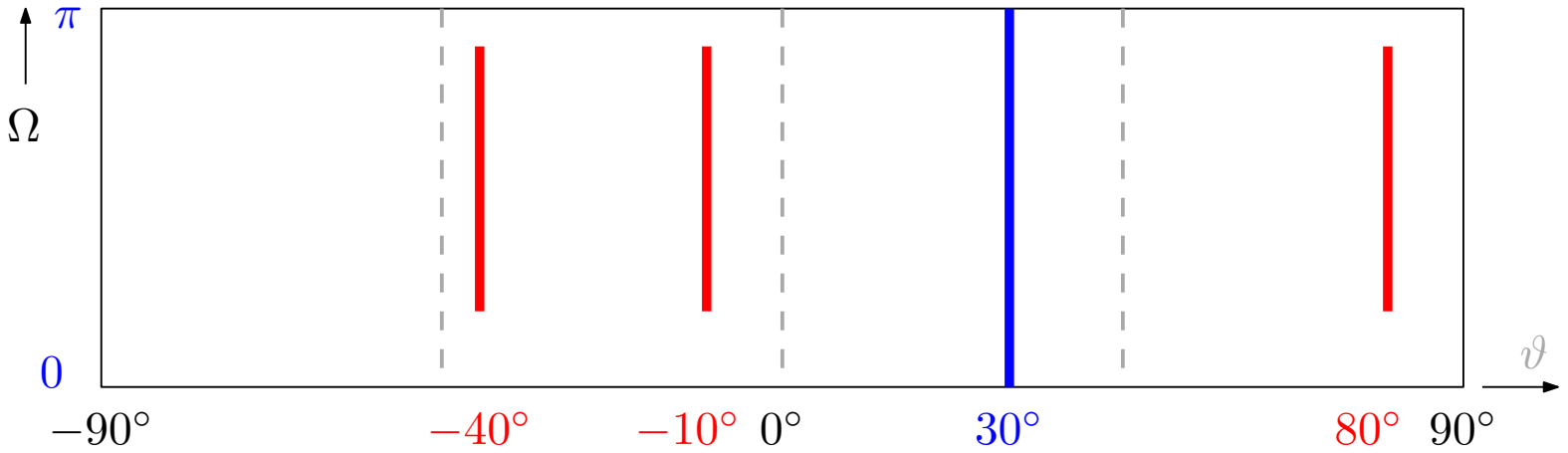
Computational Cost

- With M sensors and a TDL length of L, the complexity of a standard beamformer is dominated by the blocking matrix;
- in the proposed design, $\mathbf{w}_{\mathrm{a}} \in \mathbb{C}^{M-1}$ has degree L;
- the quiescent vector $\mathbf{w}_{\mathrm{q}}(z) \in \mathbb{C}^{M}$ has degree T;
- the blocking matrix $\boldsymbol{B}(z) \in \mathbb{C}^{(M-1)\times M}$ has degree N;
- cost comparison in multiply-accumulates (MACs):

component	GSC cost	
	polynomial	standard
quiescent beamformer	MT	ML
blocking matrix	$M(M–1)N$	$M(M–1)L^2$
adaptive filter (NLMS)	$2(M–1)L$	$2(M–1)L$

Example

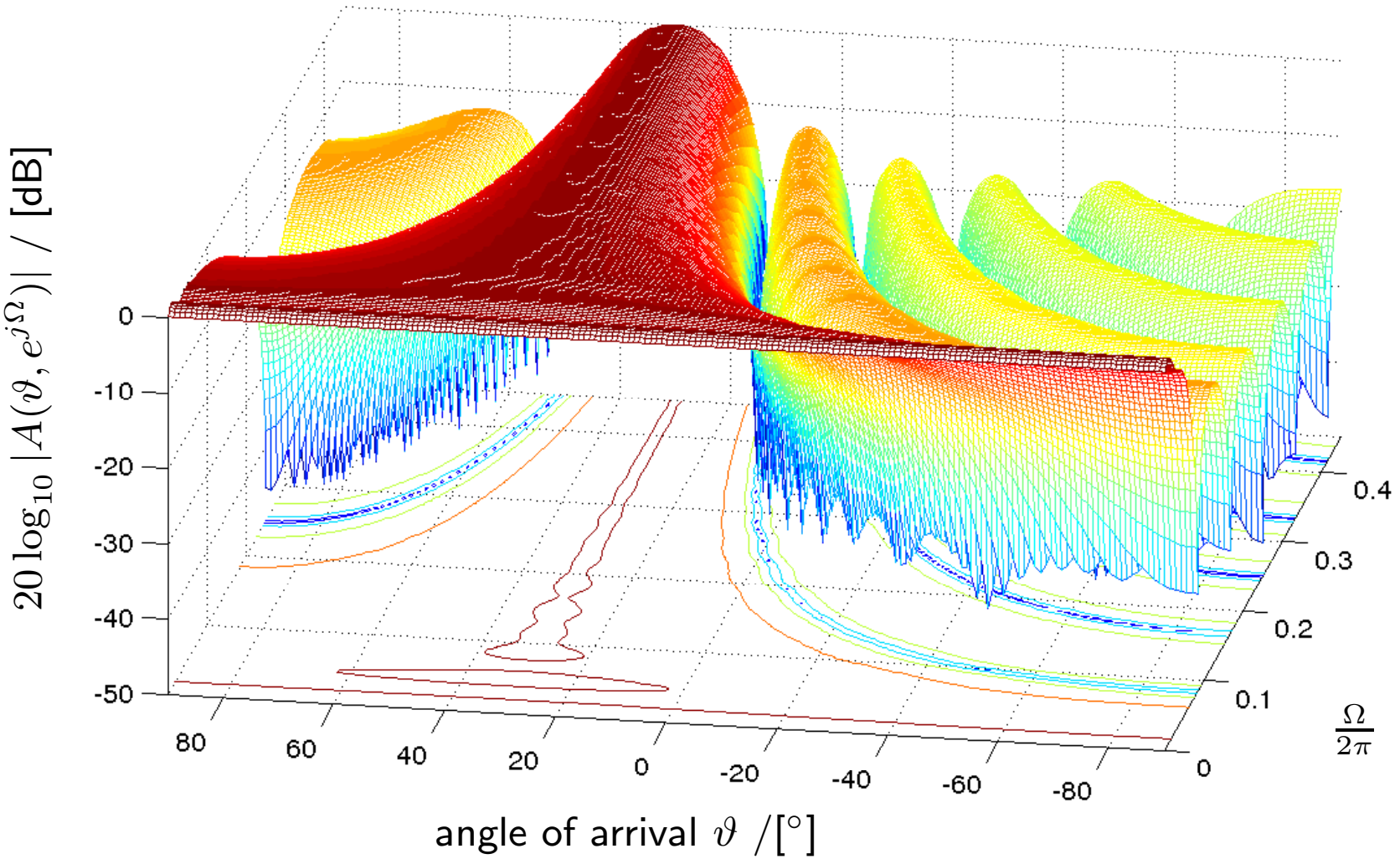
- We assume a signal of interest from $\vartheta = 30°$;
- three interferers with angles $\vartheta_i \in \{-40°, -10°, 80°\}$ active over the frequency range $\Omega = 2\pi \cdot [0.1;\ 0.45]$ at signal to interference ratio of -40 dB;



- $M = 8$ element linear uniform array is also corrupted by spatially and temporally white additive Gaussian noise at 20 dB SNR;
- parameters: $L = 175$, $T = 50$, and $N = 140$;
- cost per iteration: 10.7 kMACs (proposed) versus 1.72 MMACs (standard).

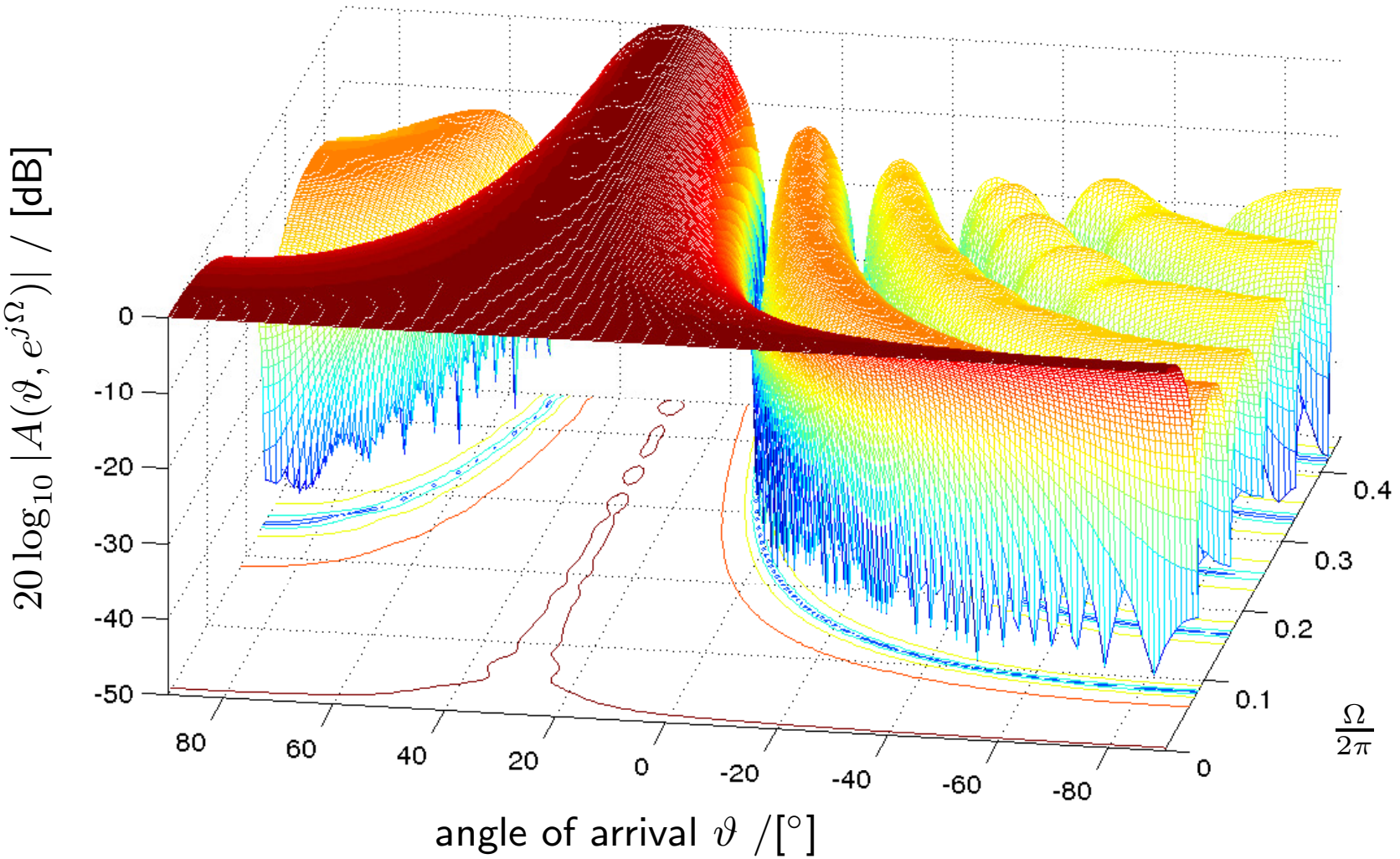
Quiescent Beamformer

- Directivity pattern of quiescent standard broadband beamformer:



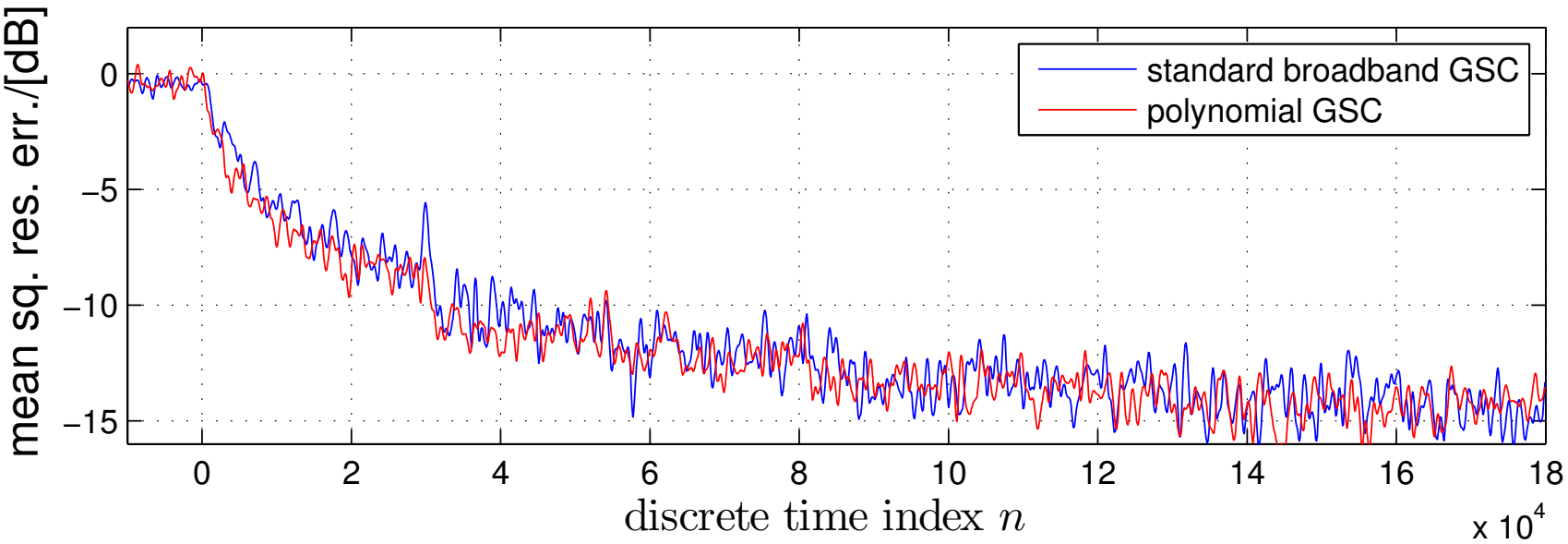
Quiescent Beamformer

- Directivity pattern of quiescent proposed broadband beamformer:



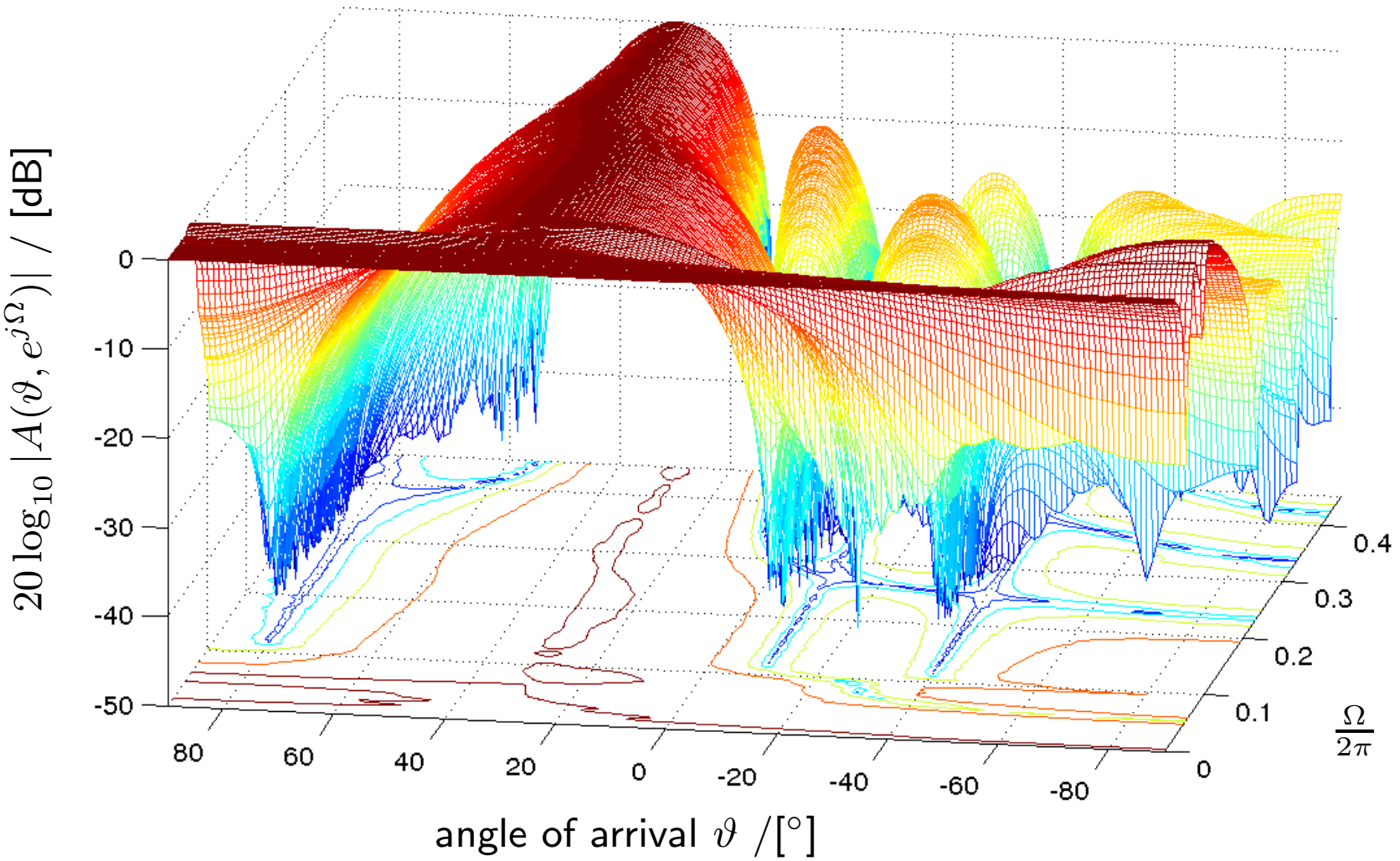
Adaptation

- Convergence curves of the two broadband beamformers, showing the residual mean squared error (i.e. beamformer output minus signal of interest):



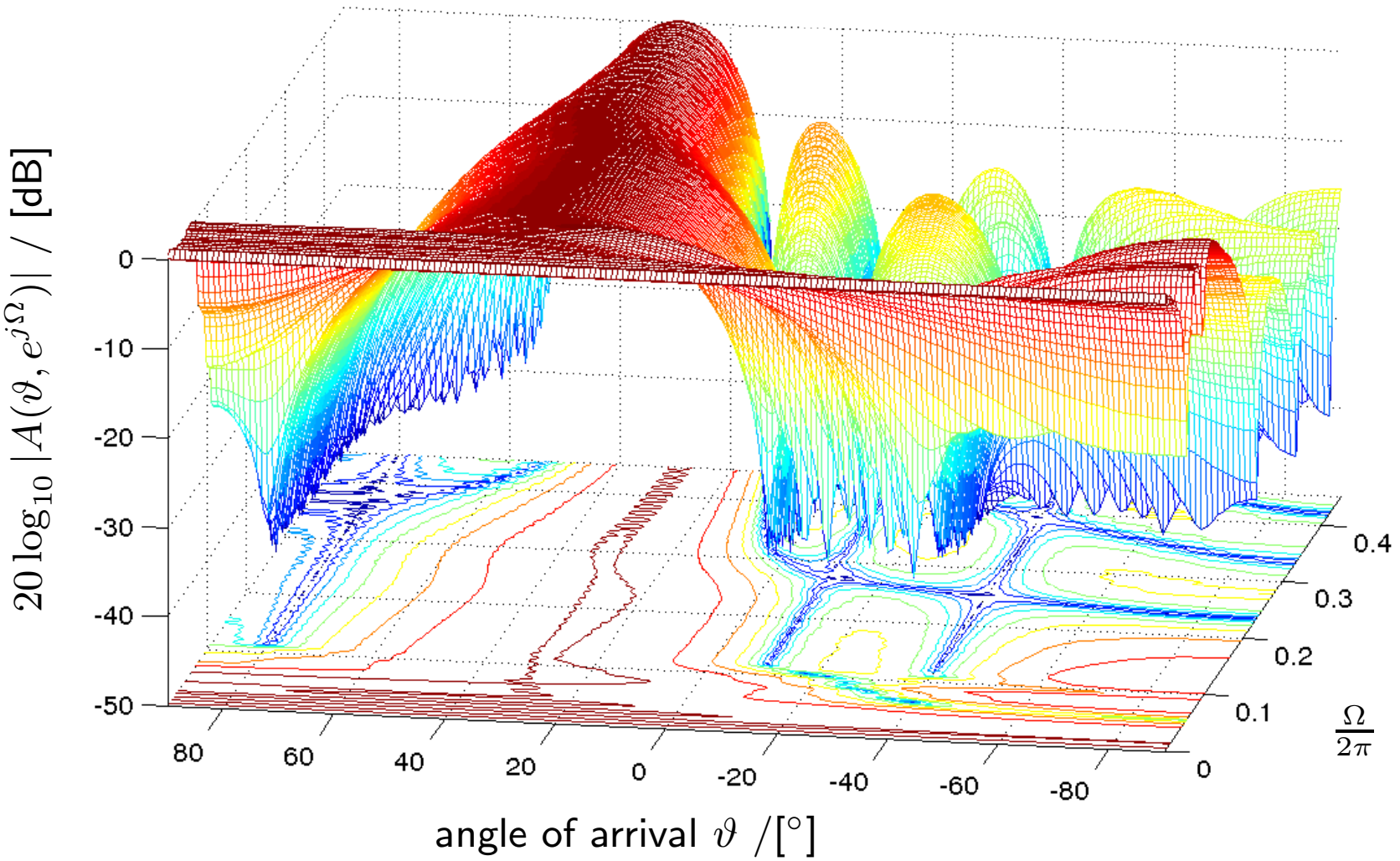
Adapted Beamformer

- Directivity pattern of adapted proposed broadband beamformer:



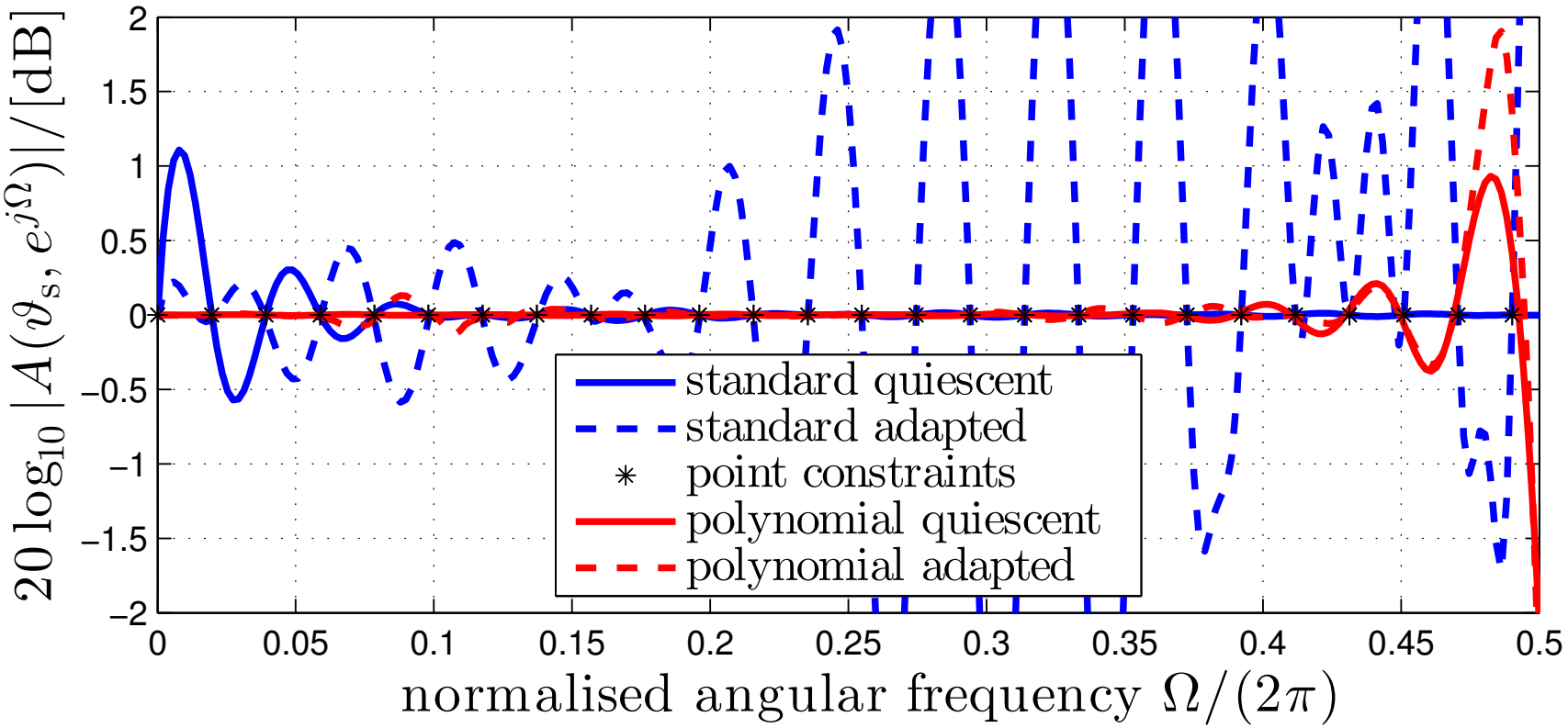
Adapted Beamformer

- Directivity pattern of adapted standard broadband beamformer:



Gain in Look Direction

- Gain in look direction $\vartheta_{\mathrm{s}} = 30^\circ$ before and after adaptation:



- due to signal leakage, the standard broadband beamformer after adaptation only maintains the point constraints but deviates elsewhere.

Broadband Beamforming Conclusions

- Based on the previous AoA estimation, beamforming can help to extract source signals and thus perform "source separation";
- broadband beamformers usually assume pre-steering such that the signal of interest lies at broadside;
- this is not always given, and difficult for arbitary array geometries;
- the proposed beamformer using a polynomial matrix formulation can implement abitrary constraints;
- the performance for such constraints is better in terms of the accuracy of the directivity pattern;
- because the proposed design decouples the complexities of the coefficient vector, the quiescent vector and block matrix, and the adaptive process, the cost is significantly lower than for a standard broadband adaptive beamformer.

Additional Material

- Papers included on the USB drives:
  1. J.G. McWhirter, P.D. Baxter, T. Cooper, S. Redif, and J. Foster: "An EVD Algorithm for Para-Hermitian Polynomial Matrices," IEEE Transactions on Signal Processing, **55**(5): 2158-2169, May 2007.
  2. S, Redif, J.G. McWhirter, and S. Weiss: "Design of FIR Paraunitary Filter Banks for Subband Coding Using a Polynomial Eigenvalue Decomposition," IEEE Transactions on Signal Processing, **59**(11): 5253-5264, Nov. 2011.
  3. P. Baxter and J.G. McWhirter: "Blind signal separation of convolutive mixtures," Proc. 37th Asilomar Conference on Signals, Systems and Computers, **1**: 124-128, November 2003.
- If interested in the discussed methods and algorithms, please download the free Matlab PEVD toolbox from

  pevd-toolbox.eee.strath.ac.uk
- for questions, please feel free to ask:
  - Stephan Weiss (stephan.weiss@strath.ac.uk) or
  - Jamie Corr (jamie.corr@strath.ac.uk).