Outline

Part I :

– Introduction
– System model and rank reduction
– Reduced-rank MMSE and MVDR designs
– Eigen-decomposition techniques
– Krylov subspace techniques
Outline (continued)

Part II:

– Joint and iterative optimization (JIO) techniques
– Joint interpolation, decimation and filtering (JIDF) techniques
– Model order selection
– Applications, perspectives and future work
– Concluding remarks
Introduction

– Reduced-rank detection and estimation techniques are fundamental tools in signal processing and communications.

– Motivation of reduced-rank signal processing:
  – robustness against noise and model uncertainties,
  – computational efficiency,
  – decompositions of signals for design and analysis,
  – inverse problems,
  – feature extraction,
  – dimensionality reduction,
  – problems with short data record, faster training.
Introduction

– Main goals of reduced-rank methods:
  – simplicity, ease of deployment,
  – to provide minimal reconstruction error losses,
  – to allow simple mapping and inverse mapping functions,
  – to improve convergence and tracking performance for dynamic signals,
  – to reduce the need for storage of coefficients,
  – to provide amenable and stable implementation,
Introduction

– Communications:
  – Interference mitigation, synchronization, fading mitigation, channel estimation.
  – Estimation with MMSE or LS criteria (Haykin [1]):

\[ w = R^{-1} p, \]

where

\( w \) is a parameter vector with \( M \) coefficients,
\( r(i) \) is the \( M \times 1 \) input data vector,
\( R = E[r(i)r^H(i)] \) is the \( M \times M \) covariance matrix,
\( p = E[d^*(i)r(i)] \) and \( d(i) \) is the desired signal.

– Detection approaches using MMSE or LS estimates.
– Problems: dimensionality of system, matrix inversions.
– How to improve performance?
– How do we deal with the computational complexity?
Introduction

- Array signal processing:
  - Beamforming, direction finding, information combining with sensors, radar and sonar (van Trees [2]).
  - Parameter estimation with the MVDR criterion:

\[ w = \xi^{-1}R^{-1}a(\Theta_k), \]

where
\[ w \] is a parameter vector with \( M \) coefficients,
\( r(i) \) is the \( M \times 1 \) input data vector,
\( R = E[r(i)r^H(i)] \) is the \( M \times M \) covariance matrix,
\( a(\Theta_k) \) is the \( M \times 1 \) array response vector and
\( \xi = a(\Theta_k)^H R^{-1} a(\Theta_k) \).

- Use of MVDR for beamforming and direction finding.
- Any idea?
- Undermodelling? → designer has to select the key features of \( r(i) \) → reduce-rank signal processing
System Model and Rank Reduction

– Consider the following linear model

\[ r(i) = H(i)s(i) + n(i), \]

where \( s(i) \) is a \( M \times 1 \) discrete-time signal organized in data vectors, \( r(i) \) is the \( M \times 1 \) input data, \( H(i) \) is a \( M \times M \) matrix and \( n(i) \) is \( M \times 1 \) noise vector.

– Dimensionality reduction → an M-dimensional space is mapped into a D-dimensional subspace.
System Model and Rank Reduction

- A general reduced-rank version of \( r(i) \) can be obtained using a transformation matrix \( S_D \) (assumed fixed here) with dimensions \( M \times D \), where \( D \) is the rank. Please see Haykin [1], Scharf-91 [3], Scharf and Tufts-87 [4], Scharf and van Veen-87[5].

- In other words, the mapping is carried out by the transformation matrix \( S_D \).

- The resulting reduced-rank observed data is given by

\[
\bar{r}(i) = S_D^H r(i)
\]

where \( \bar{r}(i) \) is a \( D \times 1 \) vector.

- **Challenge**: How to efficiently (or optimally) design \( S_D \)?
Historical Overview of Reduced-Rank Methods

– Origins of reduced-rank methods as a structured field:

  • 1987 - Louis Scharf from University of Colorado defined the problem as “a transformation in which a data vector can be represented by a reduced number of effective features and yet retain most of the intrinsic information of the input data” Scharf and Tufts-87 [4], Scharf and van Veen-87[5].

  • 1987- Scharf - Investigation and establishment of the bias versus noise variance trade-off.
Historical Overview of Reduced-Rank Methods

– Early Methods:
  ● Hotelling and Eckhart (see Scharf [3]) in the 1930’s → first methods using eigen-decompositions or principal components.
  ● Early 1990’s - applications of eigen-decomposition techniques for reduced-rank estimation in communications. See Haimovich and Bar-Ness [7], Wang and Poor [8], and Hua et al. [9].
  ● 1994 → Cai and Wang [6], Bell Labs: joint domain localised adaptive processing → radar-based scheme, medium complexity.

– Main problems of eigen-decomposition techniques:
  ● Require computationally expensive SVD or EVD, or algorithms to obtain the eigenvalues and eigenvectors.
  ● Performance degradation with the increase in the signal subspace.
Historical Overview of Reduced-Rank Methods

– 1997 - Goldstein and Reed [10], University of Southern California: cross-spectral approach.
  
  • Appropriate selection of eigenvalues values which addresses the performance degradation.
  • Remaining problem: EVD or SVD requirement.

– 1998/9 - Partial despreading (PD) of Singh and Milstein [19], University of California at San Diego:
  
  • Simple but suboptimal and restricted to CDMA multiuser detection.
Historical Overview of Reduced-Rank Methods

– Krylov subspace methods: conjugate gradient techniques developed in the 1950s.
– 1997 → Pados and Batallama [20]-[24], University of New York, Buffalo: auxiliary vector filtering (AVF) algorithm:
  • do not require SVD.
  • very fast convergence but complexity is still a problem.
– 1997 - 2004 - multistage Wiener filter (MSWF) of Goldstein, Reed and Scharf and its variants [12]-[16]:
  • State-of-the-art in the field and benchmark.
  • Very fast convergence, rank not scaling with system size.
  • equivalence between the AVF (with orthogonal AVs) and the MSWF was established by Chen, Mitra and Schniter [17].
  • Complexity is still a problem as well as the existence of numerical instability for implementation.
Historical Overview of Reduced-Rank Methods

- 2004 → de Lamare and Sampaio-Neto ( [27])- Interpolated FIR filters with time-varying interpolators: low complexity, good performance but rank limited.

- 2005 → de Lamare and Sampaio-Neto - Joint interpolation, decimation and filtering (JIDF) scheme [33]-[35] - Best known scheme, flexible, smallest complexity in the field, patented.

- 2007 → de Lamare, Haardt and Sampaio-Neto - Robust MSWF [17] - Development of a robust version of the MSWF using the constrained constant modulus (CCM) design criterion.

- 2007 → de Lamare and Sampaio-Neto - Joint iterative optimisation of filters - (JIO) - Development of a generic reduced-rank scheme that is very good for mapping and inverse mapping [28].

- 2008 → de Lamare, Sampaio-Neto and Haardt [37] - Robust JIDF-type approach called BARC - Development of a robust version of the JIDF using the CCM design criterion.
Linear MMSE Reduced-Rank Estimator Design

- The linear MMSE estimator is the vector \( w = \begin{bmatrix} w_1 & w_2 & \ldots & w_M \end{bmatrix}^T \), which is designed to minimize the MSE cost function
  
  \[
  J = E\left[|d(i) - w^H r(i)|^2\right]
  \]

  where \( d(i) \) is the desired signal.

- The solution is \( w = R^{-1} p \), where \( E[d^*(i)r(i)] \) and \( R = E[r(i)r^H(i)] \).

- The estimator \( w \) can be also be computed via adaptive algorithms, however ...

- The convergence speed and tracking of these algorithms depends on \( M \) and the eigenvalue spread. Thus, large \( M \) implies slow convergence.

- Reduced-rank schemes circumvent these limitations via a reduction in the number of coefficients and the extraction of the key features of the data.
Linear MMSE Reduced-Rank Estimator Design

– Consider a reduced-rank input vector \( \mathbf{r}(i) = S_H^D \mathbf{r}(i) \) as the input to an estimator represented by the \( D \) vector \( \mathbf{w} = [\bar{w}_1 \, \bar{w}_2 \, \ldots \, \bar{w}_D]^T \) for time interval \( i \).

– The estimator output is

\[
x(i) = \mathbf{w}^H S_H^D \mathbf{r}(i)
\]

– The MMSE design problem can be stated as

\[
\text{minimize } J(\mathbf{w}) = E[|d(i) - x(i)|^2] = E[|d(i) - \mathbf{w}^H S_H^D \mathbf{r}(i)|^2]
\]

where \( d(i) \) is the desired signal.
Linear MMSE Reduced-Rank Estimator Design

- The MMSE design with the reduced-rank parameters yields
  \[ \bar{w} = \bar{R}^{-1}\bar{p}, \]

  where
  \[ \bar{R} = E[\bar{r}(i)\bar{r}^H(i)] = S^H_D R S_D \]
  is the reduced-rank covariance matrix,
  \[ R = E[r(i)r^H(i)] \]
  is the full-rank covariance matrix,
  \[ \bar{p} = E[d^*(i)\bar{r}(i)] = S^H_D p \]
  and \[ p = E[d^*(i)r(i)] \].

- The associated MMSE for a rank \( D \) estimator is expressed by
  \[
  \text{MMSE} = \sigma^2_d - \bar{p}^H\bar{R}^{-1}\bar{p} = \sigma^2_d - p^H S_D (S^H_D R S_D)^{-1} S^H_D p
  \]
  where \( \sigma^2_d \) is the variance of \( d(i) \).
MVDR Reduced-Rank Beamformer Design

- Consider a uniform linear array (ULA) of $M$ elements.
- There are $K$ narrowband sources impinging on the array ($K < M$) with directions of arrival (DOA) $\theta_k$ for $k = 1, 2, \ldots, K$.

\[
\begin{align*}
\text{Interferer} & \quad \cdots \quad \text{Interferer} & \quad \text{SOI} \\
\text{Sample} & \quad r_1(i) & \quad r_2(i) & \quad \cdots & \quad r_M(i) \\
& \quad w_1(i) & \quad w_2(i) & \quad \cdots & \quad w_M(i) \\
\text{Adaptive} & \text{algorithm} \\
\text{Output} & \quad x(i)
\end{align*}
\]

- The received signal is given by:

\[
r(i) = \sum_{k=1}^{K} a(\theta_k) s_k(i) + n(i)
\]

- Reduced-rank array processing: The output of the array is

\[
x(i) = \bar{w}^H \bar{r}(i) = \bar{w}^H(i) S_D^H r(i)
\]
MVDR Reduced-Rank Beamformer Design

In order to design the reduced-rank beamformer \( \bar{w}(i) \) we consider the following optimization problem

\[
\text{minimize } E[|\bar{w}^H S_D^H r(i)|^2] = \bar{w}^H S_D^H R S_D \bar{w}
\]

subject to \( \bar{w}^H S_D^H a(\theta_k) = 1 \)

Approach to obtain a solution : method of Lagrange multipliers

\[
\mathcal{L}(\bar{w}, \lambda) = E[|\bar{w}^H S_D^H r(i)|^2] + \lambda (\bar{w}^H S_D^H a(\theta_k) - 1)
\]

The solution to this design problem is

\[
\bar{w} = \frac{(S_D^H R S_D)^{-1} S_D^H a(\theta_k)}{a^H(\theta_k) S_D(i) (S_D^H R S_D)^{-1} S_D^H a(\theta_k)} = \frac{\bar{R}^{-1} \bar{a}(\theta_k)}{\bar{a}^H(\theta_k) \bar{R}^{-1} \bar{a}(\theta_k)}
\]

where the reduced-rank covariance matrix is \( \bar{R} = E[\bar{r}(i) \bar{r}^H(i)] = S_D^H R S_D \) and the reduced-rank steering vector is \( \bar{a}(\theta_k) = S_D^H a(\theta_k) \).
MVDR Reduced-Rank Beamformer Design

- The associated minimum variance (MV) for an MVDR beamformer with rank \(D\) is

\[
MV = \frac{1}{\bar{a}(\theta_k)^H \bar{R}^{-1} \bar{a}(\theta_k)}
\]

\[
= \frac{1}{\bar{a}(\theta_k)^H S_D (S_D^H R S_D)^{-1} S_D^H a(\theta_k)}
\]

- The above expression can be used for direction finding by replacing the angles \(\theta_k\) with a time-varying parameter \((\omega)\) in order to scan the possible angles.

- It can also be employed for general applications of spectral estimation including spectral sensing.
Eigen-Decomposition Techniques

– Why are eigen-decomposition techniques used?
– For MMSE parameter estimation and a rank $D$ estimator we have

$$\text{MMSE} = \sigma_d^2 - \mathbf{p}^H \mathbf{S}_D \left( \mathbf{S}_D^H \mathbf{R} \mathbf{S}_D \right)^{-1} \mathbf{S}_D^H \mathbf{p}$$

– Taking the gradient of MMSE with respect to $\mathbf{S}_D$, we get

$$\mathbf{S}_{D,\text{opt}} = [v_1 \ldots v_D],$$

where $v_d$ for $d = 1, \ldots, D$ are the eigenvectors of $\mathbf{R}$.

– For MV parameter estimation and a rank $D$ estimator we have

$$\text{MV} = \frac{1}{\mathbf{a}(\theta_k)^H \mathbf{S}_D \left( \mathbf{S}_D^H \mathbf{R} \mathbf{S}_D \right)^{-1} \mathbf{S}_D^H \mathbf{a}(\theta_k)}$$

– Taking the gradient of MV with respect to $\mathbf{S}_D$, we get

$$\mathbf{S}_{D,\text{opt}} = [v_1 \ldots v_D]$$
Eigen- Decomposition Techniques

- Rank reduction is accomplished by eigen-decomposition on the input data covariance matrix

\[ R = V \Lambda V^H, \]

where

- Early techniques: selection of eigenvectors \( v_j \) (\( j = 1, \ldots, M \)) corresponding to the largest eigenvalues \( \lambda_j \)
- Transformation matrix is

\[ S_D(i) = [v_1 \ldots v_D] \]
Eigen-Decomposition Techniques

– Cross-spectral approach of Goldstein and Reed: choose eigenvectors that minimise the design criterion
  → Transformation matrix is
  \[ S_D(i) = [v_i \ldots v_t] \]

– Problems: Complexity \( O(M^3) \), optimality implies knowledge of \( R \) but this has to be estimated.

– Complexity reduction: adaptive subspace tracking algorithms (popular in the end of the 90s) but still complex and susceptible to tracking problems.

– Can we skip or circumvent an eigen-decomposition?
Krylov Subspace Techniques

- Krylov subspace techniques have a rich history in solving systems of equations and in numerical linear algebra.

- In array signal processing and communications, we are usually interested in solving symmetric and positive definite systems.

- In this context, one of the most important Krylov subspace methods is the conjugate gradient technique invented by Hestenes and Stiefel in 1950s.

- **Main idea**: to solve $Rw = p$ in the Krylov subspace that spans after $k$ iterations span\{\(p, Rp, \ldots, R^{k-1}p\}\).

- Complexity : quadratic in $M$. 

Conjugate Gradient Techniques

- **Main idea**: to solve $Rw = p$ in $k$ iterations.
  Initialize all parameter vectors $g_0 = p - Rw_0$, $w_0$, $v_0$
  For $k = 1, \ldots, K$ do:
  - Calculate step size:
    $$\alpha_k = \frac{g_{k-1}^H g_{k-1}}{p_k^H R p_k}$$
  - Compute parameters:
    $$w_k = w_{k-1} + \alpha v_k$$
  - Calculate step size:
    $$\beta_k = \frac{v_{k-1}^H R g_{k-1}}{v_{k-1}^H R v_{k-1}}$$
  - Compute direction vectors:
    $$v_k = g_k + \beta_k v_{k-1}$$
  - Calculate negative gradient:
    $$g_k = g_{k-1} - \alpha_k R v_k$$
Multi-stage Wiener Filter

- Rank reduction is accomplished by a successive refinement procedure that generates a set of basis vectors, i.e. the signal subspace, known in numerical analysis as the Krylov subspace.

- Design: use of nested filters $c_j$ ($j = 1, \ldots, M$) and blocking matrices $B_j$ for the decomposition $\rightarrow$ Projection matrix is

$$S_D(i) = [p, R_p, \ldots, R^{D-1}p]$$

- Advantages: rank $D$ does not scale with system size, very fast convergence.
- Problems: complexity slightly inferior to RLS algorithms, not robust to signature mismatches in blind operation.
Robust Multi-stage Wiener Filter

- Rank reduction is accomplished by a similar successive refinement procedure to original MSWF. However, the design is based on the CCM criterion (de Lamare, Haardt and Sampaio-Neto, IEEE TSP, 2008).

- Transformation matrix:

\[
S_D(i) = \begin{bmatrix} q(i), R(i)q(i), \ldots, R^{D-1}(i)q(i) \end{bmatrix}
\]

- The reduced-rank CCM parameter vector with rank \(D\) is

\[
\bar{w}(i + 1) = \left( S_D^H(i)R(i)S_D(i) \right)^{-1} S_D^H(i)q(i),
\]

where

\[
q(i) = d(i) - (p^H(i)R^{-1}(i)p(i))^{-1}(p^H(i)R^{-1}(i)d(i) - \nu)p(i),
\]

\[
d(i) = E[x^*(i)S_D^H(i)r(i)]
\]
Applications: Interference Suppression for CDMA

- We assess BER performance of the supervised LS, the CMV-LS and the CCM-LS and their full-rank and reduced-rank versions.

- The DS-CDMA system uses random sequences with $N = 64$.

- We use 3-path channels with powers $p_{k,l}$ given by 0, $-3$ and $-6$ dB. In each run the spacing between paths is obtained from a discrete uniform random variable between 1 and 2 chips.

- Power distribution amongst the users: Follows a log-normal distribution with associated standard deviation of 1.5 dB.

- All LS type estimators use $\lambda = 0.998$ to ensure good performance and all experiments are averaged over 200 runs.
Applications: Interference Suppression for CDMA

- BER convergence performance at $E_b/N_0 = 12$ dB.

$N=64$, $f_dT=0.0001$, $K=16$ (i=1–1500) users, $K=24$ (i=1501–3000) users

![Graph showing BER vs. Number of received symbols for different methods.]
JIO Techniques

- Rank reduction is performed by joint and iterative optimisation (JIO) of the rank-reduction matrix $S_D(i)$ and reduced-rank estimator $\bar{w}(i)$.

- Design criteria: MMSE, LS, LCMV, etc.
- Adaptive algorithms: LMS, RLS, etc.
- Highlights: rank $D$ does not scale with system size, very fast convergence, proof of global convergence established, very simple.
MMSE Design of JIO Scheme

- The MMSE expressions for the filters $S_D(i)$ and $\bar{w}(i)$ can be computed through the following cost function:

$$J = E\left[|d(i) - \bar{w}^H(i)S_D^H(i)r(i)|^2\right]$$

- By fixing $S_D(i)$ and minimizing the cost function with respect to $\bar{w}(i)$, the reduced-rank estimator becomes

$$\bar{w}(i) = \bar{R}^{-1}(i)\bar{p}(i)$$

where

$$\bar{R}(i) = E[S_D^H(i)r(i)r^H(i)S_D(i)] = E[\bar{r}(i)\bar{r}^H(i)],$$

$$\bar{p}(i) = E[d^*(i)S_D^H(i)r(i)] = E[d^*(i)\bar{r}(i)].$$
MMSE Design of JIO Scheme

– Fixing $\bar{w}(i)$ and minimizing the cost function with respect to $S_D(i)$, we get

$$S_D(i) = R^{-1}(i)P_D(i)R_w^{-1}(i)$$

where

$R(i) = E[r(i)r^H(i)]$,

$P_D(i) = E[d^*(i)r(i)\bar{w}^H(i)]$ and

$R_w(i) = E[\bar{w}(i)\bar{w}^H(i)]$.

– The associated MMSE is

$$\text{MMSE} = \sigma_d^2 - \bar{p}^H(i)\bar{R}^{-1}(i)\bar{p}(i)$$

where $\sigma_d^2 = E[|d(i)|^2]$. 
MMSE Design of JIO Scheme

- The filter expressions for $\tilde{w}(i)$ and $S_D(i)$ are functions of one another and thus it is necessary to iterate them with an initial guess to obtain a solution.

- Unlike prior art, the JIO scheme provides an iterative exchange of information between the reduced-rank estimator and the transformation matrix.

- The key strategy lies in the joint optimization of the filters → the method is guided by the optimization algorithm.

- The rank $D$ or model order must be set by the designer to ensure appropriate or adjusted on-line.
Adaptive JIO-LMS Algorithm

Initialize all parameter vectors, dimensions

For each data vector $i = 1, \ldots, Q$ do:

- Perform dimensionality reduction:
  \[
  \bar{r}(i) = S_D^H(i) r(i)
  \]

- Estimate parameters
  \[
  S_D(i + 1) = S_D(i) + \eta(i)e^*(i)r(i)\bar{w}^H(i)
  \]
  \[
  \bar{w}(i + 1) = \bar{w}(i) + \mu(i)e^*(i)\bar{r}(i)
  \]

where $e(i) = d(i) - \bar{w}^H(i)S_D^H(i)r(i)$. 
Applications : Interference Suppression for CDMA

– We consider the uplink of a symbol synchronous BPSK DS-CDMA system with $K$ users, $N$ chips per symbol and $L$ propagation paths.

– Initialization : for all simulations, we use $\bar{w}(0) = 0_{D,1}$, $S_D(0) = \begin{bmatrix} I_D & 0_{D,M-D} \end{bmatrix}^T$.

– We assume $L = 9$ as an upper bound on the channel delay spread, use 3-path channels with relative powers given by 0, −3 and −6 dB, where in each run the spacing between paths is obtained from a discrete uniform random variable between 1 and 2 chips and average the experiments over 200 runs.

– The system has a power distribution amongst the users for each run that follows a log-normal distribution with associated standard deviation equal to 1.5 dB.
Applications: Interference Suppression for CDMA

N=64, K=20 users, E_b/N_0=12 dB, Data Record = 500 symbols

<table>
<thead>
<tr>
<th>Rank (D)</th>
<th>SINR (dB)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Full-Rank-NLMS</td>
<td></td>
</tr>
<tr>
<td>Full-Rank-RLS</td>
<td></td>
</tr>
<tr>
<td>MWF-NLMS</td>
<td></td>
</tr>
<tr>
<td>MWF-RLS</td>
<td></td>
</tr>
<tr>
<td>AVF</td>
<td></td>
</tr>
<tr>
<td>Proposed-NLMS</td>
<td></td>
</tr>
<tr>
<td>MMSE</td>
<td></td>
</tr>
</tbody>
</table>
Applications: Interference Suppression for CDMA

N=64, K=20 users, $E_b/N_0=12$ dB

Number of received symbols vs. SINR (dB)

- Full-Rank-NLMS ($\mu_0=0.1$)
- Full-Rank-RLS ($\lambda=0.998$)
- MWF-NLMS ($D=3, \mu_0=0.1$)
- MWF-RLS ($D=6, \lambda=0.998$)
- AVF ($D=8, \lambda=0.998$)
- Proposed-NLMS ($D=4, \mu_0=0.05, \eta_0=0.1$)
- MMSE
Applications: Interference Suppression for CDMA

N=64, K=16 users, $E_b/N_0=15$ dB, $f_d T=0.0001$

Number of received symbols

BER

Full-Rank-NLMS ($\mu_0=0.075$)
Full-Rank-RLS ($\lambda=0.998$)
MWF-NLMS ($D=3, \mu_0=0.075$)
MWF-RLS ($D=6, \lambda=0.998$)
AVF ($D=6, \lambda=0.998$)
Proposed-NLMS ($D=4, \mu_0=0.05, \eta_0=0.075$)
MVDR Design of JIO Scheme

– Main differences in approach: the beamformers $S_D(i)$ and $\bar{w}(i)$ are jointly optimized and certain key quantities are assumed statistically independent.

– The MVDR expressions for the beamformers $S_D(i)$ and $\bar{w}(i)$ can be computed via the proposed optimization problem

$$\begin{align*}
\text{minimize} & \quad E\left[ |\bar{w}^H(i)S_D^H(i)r(i)|^2 \right] = \bar{w}^H(i)S_D^H(i)R_{SD}(i)\bar{w}(i) \\
\text{subject to} & \quad \bar{w}^H(i)S_D^H(i)a(\theta_k) = 1
\end{align*}$$

– Solution → method of Lagrange multipliers

$$\mathcal{L}(S_D(i), \bar{w}(i), \lambda) = E\left[ |\bar{w}^H(i)S_D^H(i)r(i)|^2 \right] + \lambda(\bar{w}^H(i)S_D^H(i)a(\theta_k) - 1)$$
MVDR Design of JIO Scheme

– By fixing \( \tilde{w}(i) \), minimizing \( \mathcal{L}(S_D(i), \tilde{w}(i), \lambda) \) with respect to \( S_D(i) \) and solving for \( \lambda \), we get

\[
S_D(i) = \frac{R^{-1}a(\theta_k)\tilde{w}^H(i)R_w^{-1}}{\tilde{w}^H(i)R_w^{-1}\tilde{w}(i)a^H(\theta_k)R^{-1}a(\theta_k)},
\]

where

\[
R = E[r(i)r^H(i)] \quad \text{and} \quad R_w = E[\tilde{w}(i)\tilde{w}^H(i)].
\]

– A simplified expression for \( S_D(i) \) obtained analytically with the exploitation of the constraint is given by

\[
S_D(i) = \frac{P(i)a(\theta_k)\bar{a}^H(\theta_k)}{a^H(\theta_k)P(i)a(\theta_k)}
\]
MVDR Design of JIO Scheme

- By fixing $S_D(i)$, minimizing the Lagrangian with respect to $\bar{w}(i)$ and solving for $\lambda$, we arrive at the expression for $\bar{w}(i)$

$$\bar{w}(i) = \frac{\bar{R}^{-1}(i)\bar{a}(\theta_k)}{\bar{a}^H(\theta_k)\bar{R}^{-1}(i)\bar{a}(\theta_k)},$$

where

$$\bar{R}(i) = S_D^H(i)E[r(i)r^H(i)]S_D(i) = E[\bar{r}(i)\bar{r}^H(i)],$$

$$\bar{a}(\theta_k) = S_D^H(i)a(\theta_k).$$

- The associated MV is

$$\text{MV} = \frac{1}{\bar{a}^H(\theta_k)\bar{R}^{-1}(i)\bar{a}(\theta_k)}$$
MVDR Design of JIO Scheme

- The expressions of the beamformers $\bar{w}(i)$ and $S_D(i)$ are not closed-form solutions.
- They are functions of each other. Therefore, it is necessary to iterate the expressions with initial values to obtain a solution.
- Existence of multiple solutions (which are identical with respect to the MMSE and symmetrical).
- Global convergence to the optimal reduced-rank LCMV filter (eigen-decomposition with known covariance matrix) has been established.
- The key strategy lies in the joint optimization of the filters.
- The rank $D$ must be adjusted by the designer to ensure appropriate performance or can be estimated via another algorithm.
Adaptive MVDR-LMS Algorithm

Initialize all parameter vectors, dimensions

For each data vector $i = 1, \ldots, Q$ do:

- Perform dimensionality reduction:
  $$\tilde{r}(i) = S_D^H(i) r(i)$$

- Estimate parameters

$$S_D(i + 1) = S_D(i) - \mu_s x^*(i) \left[ r(i) \tilde{w}^H(i) - a(\theta_k) \tilde{w}^H(i) a^H(\theta_k) r(i) \right]$$

$$\tilde{w}(i + 1) = \tilde{w}(i) - \mu_w x^*(i) \left[ I - \left( \bar{a}^H(\theta_k) \bar{a}(\theta_k) \right)^{-1} \bar{a}(\theta_k) \bar{a}^H(\theta_k) \right] \tilde{r}(i)$$
# Complexity of MVDR-JIO ALgorithms

<table>
<thead>
<tr>
<th>Algorithm</th>
<th>Additions</th>
<th>Multiplications</th>
</tr>
</thead>
<tbody>
<tr>
<td>Full-rank-SG [1]</td>
<td>$3M + 1$</td>
<td>$3M + 2$</td>
</tr>
<tr>
<td>Full-rank-RLS [1]</td>
<td>$3M^2 - 2M + 3$</td>
<td>$6M^2 + 2M + 2$</td>
</tr>
<tr>
<td>JIO-LMS</td>
<td>$3DM + 2M$</td>
<td>$3DM + M$</td>
</tr>
<tr>
<td></td>
<td>$+2D - 2$</td>
<td>$+5D + 2$</td>
</tr>
<tr>
<td>JIO-RLS</td>
<td>$3M^2 - 2M + 3$</td>
<td>$7M^2 + 2M$</td>
</tr>
<tr>
<td></td>
<td>$+3D^2 - 8D + 3$</td>
<td>$+7D^2 + 9D$</td>
</tr>
<tr>
<td></td>
<td>$+3D - 2$</td>
<td>$+2DM + 4D + 1$</td>
</tr>
<tr>
<td>MSWF-RLS [12]</td>
<td>$DM^2 + M^2 + 6D^2$</td>
<td>$DM^2 + M^2$</td>
</tr>
<tr>
<td></td>
<td>$-8D + 2$</td>
<td>$+2DM + 3D + 2$</td>
</tr>
<tr>
<td>AVF [24]</td>
<td>$D((M)^2 + 3(M - 1)^2) - 1$</td>
<td>$D(4M^2 + 4M + 1)$</td>
</tr>
<tr>
<td></td>
<td>$+D(5(M - 1) + 1) + 2M$</td>
<td>$+4M + 2$</td>
</tr>
</tbody>
</table>
Complexity of JIO-MVDR Algorithms

![Graph showing the complexity comparison of different algorithms for D=6. The graph plots the number of additions and multiplications against the number of sensors (M). The algorithms compared are Full-rank-SG, Full-rank-RLS, Proposed-SG, Proposed-RLS, MSWF-SG, MSWF-RLS, and AVF.](image_url)
Applications: MVDR Beamforming

- A smart antenna system with a ULA containing $M$ sensor elements and half wavelength inter-element spacing is considered.

- Figure of merit: the SINR, which is defined as

$$\text{SINR}(i) = \frac{\bar{w}^H(i)S_D^H(i)R_s(i)S_D(i)\bar{w}(i)}{\bar{w}^H(i)S_D^H(i)R_I(i)S_D(i)\bar{w}(i)}$$

- The signal-to-noise ratio (SNR) is defined as $\text{SNR} = \frac{\sigma_d^2}{\sigma^2}$.

- Initialization: $\bar{w}(0) = [1 \ 0 \ \ldots \ 0]$ and $S_D(0) = [I_D^T \ 0_D^{T \times (M-D)}]$, where $0_D^{D \times M-D}$ is a $D \times (M - D)$ matrix with zeros in all experiments.
Applications: MVDR Beamforming

M=32, K=8 users, SNR=15 dB, data record: 250 snapshots

<table>
<thead>
<tr>
<th>Rank (D)</th>
<th>SINR (dB)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
</tr>
</tbody>
</table>

- Full-Rank-SG
- Full-Rank-RLS
- MSWF-SG
- MSWF-RLS
- AVF
- Proposed-SG
- Proposed-RLS
- LCMV-Optimal

Graph showing SINR (dB) vs. Rank (D) for various algorithms.
Applications: MVDR Beamforming

M=32, SNR=15 dB

Number of snapshots

SINR (dB)

Full-Rank-SG ($\mu=0.001$)
Full-Rank-RLS
MSWF-SG ($D=3, \mu=0.001$)
MSWF-RLS ($D=6, \lambda=0.998$)
AVF ($D=8, \lambda=0.998$)
Proposed-SG ($D=4, \mu_s=0.001, \mu_w=0.0001$)
Proposed-RLS ($D=5, \lambda=0.998$)
LCMV-Optimal
Applications: Direction of Arrival Estimation

- A smart antenna system with a ULA containing $M$ sensor elements and half wavelength inter-element spacing is considered.

- We compare the proposed LCMV JIO method with an LS algorithm with the Capon, MUSIC, ESPRIT, AVF, and CG methods, and run $K = 1000$ iterations to get each curve.

- The spatial smoothing (SS) technique is employed for each algorithm to improve the performance in the presence of correlated sources.

- The DOAs are considered to be resolved if $|\hat{\theta}_{JISO} - \theta_k| < 1^\circ$.

- The probability of resolution is used as a figure of merit and plotted against the number of snapshots.
Applications: Direction of Arrival Estimation

Parameters: Probability of resolution versus number of snapshots (separation $3^\circ$, SNR = $-2$dB, $q = 2$, $c = 0.9$, $m = 30$, $r = 6$, $\delta = 5 \times 10^{-4}$, $\alpha = 0.998$, $n = 26$)
Applications: Direction of Arrival Estimation

Parameters: Probability of resolution versus number of snapshots (separation $3^\circ$, SNR= $-5$dB, $q=10$, $m=50$, $r=6$, $\delta = 5 \times 10^{-4}$, $\alpha = 0.998$, $n = 41$)
Applications: Direction of Arrival Estimation

Parameters: Probability of resolution versus snapshots (separation $3^\circ$, SNR= 0dB, $q_w = 9$, $m = 50$, $r = 6$, $\delta = 5 \times 10^{-4}$, $\alpha = 0.998$, $n = 41$). We assume an incorrect number of sources $q_w = 9$ instead of $q = 10$. 

![Graph showing probability of resolution versus number of snapshots for different algorithms. The graph includes lines for AV, CG, Capon, ESPRIT, MUSIC, JISO, Capon-SS, ESPRIT-SS, MUSIC-SS, and JISO-SS. The x-axis represents the number of snapshots, ranging from 10 to 100. The y-axis represents the probability of resolution, ranging from 0 to 1. The graph shows how the probability of resolution changes with the number of snapshots for each algorithm.]
- Interpolated received vector: \( r_I(i) = V^H(i)r(i) \)
- Decimated received vector for branch \( b \): \( \bar{r}(i) = D_b(i)V^H(i)r(i) \)
- Selection of decimation branch \( D(i) \): Euclidean distance
- Expression of estimate as a function of \( v(i), D(i) \) and \( w(i) \):
  \[
  x(i) = \bar{w}^H(i)S_D^H(i)r(i) = \bar{w}^H(i)D_b(i)V^H(i)r(i) = \bar{w}^H(i)D(i)\mathbb{R}_o(i)
  \]
- Joint optimisation of \( v(i), D(i) \) and \( \bar{w}(i) \)
JIDF Techniques

– **Decimation schemes**: Optimal, uniform, random, pre-stored.
– The decimation pattern \( D(i) \) is selected according to:

\[
D(i) = D_b \quad \text{when} \quad D_b(i) = \arg \min_{1 \leq b \leq B} |e_b(i)|^2
\]

– Optimal decimator: combinatorial problem with \( B \) possibilities

\[
B = M \cdot (M - 1) \ldots (M - M/L + 1) = \frac{M!}{(M - M/L)!}
\]

\( \frac{M!}{(M - M/L)!} \) terms

– Suboptimal decimation schemes:

– Uniform (U) Decimation

– Pre-Stored (PS) Decimation.

– Random (R) Decimation.
JIDF Techniques
– General framework for decimation schemes

\[ D_b = \begin{bmatrix}
0 & \ldots & 0 & 1 & 0 & 0 & 0 & 0 & \ldots & 0 & 0 & 0 \\
\underbrace{0 \ldots 0}_{r_1 \text{ zeros}} & 1 & 0 & 0 & 0 & \ldots & 0 & 0 & 0 & \ldots & \ldots & \ldots \\
\vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\
0 & \ldots & 0 & 1 & 0 & 0 & 0 & \ldots & 0 & 0 & 0 & 0 \\
\underbrace{0 \ldots 0}_{r_m \text{ zeros}} & 1 & 0 & 0 & 0 & \ldots & 0 & 0 & 0 & \ldots & \ldots & 0 \\
\vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \ldots & 0 & 1 \\
\underbrace{0 \ldots 0}_{r_D \text{ zeros}} & 1 & 0 & 0 & \ldots & 0 & 0 & 0 & 0 & \ldots & \ldots & 0 \\
\underbrace{0 \ldots 0}_{(M-r_D-1) \text{ zeros}} & 1 & 0 & 0 & \ldots & 0
\end{bmatrix}
\]

where \( m \) (\( m = 1, 2, \ldots, M/L \)) denotes the \( m \)-th row and \( r_m \) is the number of zeros given by the decimation strategy.

– Suboptimal decimation schemes:

a. Uniform (U) Decimation with \( B = 1 \rightarrow r_m = (m - 1)L \).

b. Pre-Stored (PS) Decimation. We select \( r_m = (m - 1)L + (b - 1) \) which corresponds to the utilization of uniform decimation for each branch \( b \) out of \( B \) branches.

c. Random (R) Decimation. We choose \( r_m \) as a discrete uniform random variable between 0 and \( M - 1 \).
Linear MMSE Design of Parameter Vectors

- The MMSE expressions for \( \bar{w}(i) \) and \( \mathbf{v}(i) \) can be computed via the minimization of the cost function

\[
J_{\text{MSE}}^{(\mathbf{v}(i), \mathbf{D}(i), \bar{w}(i))} = E[|d(i) - \mathbf{v}^H(i) \mathbf{R}_o^T(i) \mathbf{D}^T(i) \bar{w}^*(i)|^2]
\]

- Fixing the interpolator \( \mathbf{v}(i) \) and minimizing the cost function with respect to \( \bar{w}(i) \) the interpolated Wiener filter weight vector is

\[
\bar{w}(i) = \alpha(\mathbf{v}) = \bar{\mathbf{R}}^{-1}(i) \bar{\mathbf{p}}(i)
\]

where

\[
\bar{\mathbf{R}}(i) = E[\bar{\mathbf{r}}(i)\bar{\mathbf{r}}^H(i)],
\]

\[
\bar{\mathbf{p}}(i) = E[d^*(i)\bar{\mathbf{r}}(i)],
\]

\[
\bar{\mathbf{r}}(i) = \mathbf{R}(i)\mathbf{v}^*(i).
\]
Linear MMSE Design of Parameter Vectors

- Fixing $\bar{w}(i)$ and minimizing the cost function with respect to $v(i)$ the interpolator weight vector is

$$v(i) = \beta(\bar{w}) = R_u^{-1}(i)p_u(i)$$

where

$$R_u(i) = E[u(i)u^H(i)], \quad p_u(i) = E[d^*(i)u(i)] \quad \text{and} \quad u(i) = R^T(i)\bar{w}^*(i)$$

- The associated MSE expressions are

$$J(v) = J_{MSE}(\alpha(v), v) = \sigma_d^2 - \bar{p}^H(i)R^{-1}(i)\bar{p}(i)$$

$$J_{MSE}(\bar{w}, \beta(\bar{w})) = \sigma_d^2 - p_u^H(i)R_u^{-1}(i)p_u(i)$$

where $\sigma_d^2 = E[|d(i)|^2]$.

- The points of global minimum can be obtained by $v_{opt} = \arg\min_v J(v)$ and $\bar{w}_{opt} = \alpha(v_{opt})$ or $\bar{w}_{opt} = \arg\min_{\bar{w}} J_{MSE}(\bar{w}, \beta(\bar{w}))$ and $v_{opt} = \beta(\bar{w}_{opt})$. 
Initialize all parameter vectors, dimensions, number of branches $B$ and select decimation technique.

For each data vector $i = 1, \ldots, Q$ do:

- Select decimation branch that minimizes $e_b(i) = d(i) - \mathbf{w}^H(i)\bar{r}(i)$
- Make $\bar{r}(i) = \bar{r}_b(i)$ when $b = \arg \min_{1 \leq b \leq B} |e_b(i)|^2$
- Estimate parameters

$$v(i + 1) = v(i) + \eta e^*(i)u(i)$$

$$\bar{w}(i + 1) = \bar{w}(i) + \mu e^*(i)\bar{r}(i)$$

where $u(i) = R^T(i)\bar{w}^*(i)$ and $\bar{r}(i) = D(i)V^H(i)r(i)$. 

Adaptive JIDF-LMS Algorithms
## Complexity of JIDF Algorithms

<table>
<thead>
<tr>
<th>Algorithm</th>
<th>Number of operations per symbol</th>
<th>Additions</th>
<th>Multiplications</th>
</tr>
</thead>
<tbody>
<tr>
<td>Full-rank-LMS</td>
<td></td>
<td>$2M$</td>
<td>$2M + 1$</td>
</tr>
<tr>
<td>Full-rank-RLS</td>
<td></td>
<td>$3(M - 1)^2 + M^2 + 2M$</td>
<td>$6M^2 + 2M + 2$</td>
</tr>
<tr>
<td>JIDF-LMS</td>
<td></td>
<td>$(B + 1)(D) + 2N_I$</td>
<td>$(B + 2)D$</td>
</tr>
<tr>
<td>JIDF-RLS</td>
<td></td>
<td>$3(D - 1)^2 + 3(N_I - 1)^2 + (D - 1)N_I + N_IM + (D)^2 + N_I^2 + (B + 1)D + 2N_I$</td>
<td>$6(D)^2 + 6N_I^2 + DN_I + 2$</td>
</tr>
<tr>
<td>MWF-LMS</td>
<td></td>
<td>$D(2(\bar{M} - 1)^2 + \bar{M} + 3)$</td>
<td>$D(2\bar{M}^2 + 5\bar{M} + 7)$</td>
</tr>
<tr>
<td>MWF-RLS</td>
<td></td>
<td>$D(4(\bar{M} - 1)^2 + 2\bar{M})$</td>
<td>$D(4\bar{M}^2 + 2\bar{M} + 3)$</td>
</tr>
<tr>
<td>AVF</td>
<td></td>
<td>$D((M)^2 + 3(M - 1)^2) - 1 + D(5(M - 1) + 1) + 2M$</td>
<td>$D(4(M)^2 + 4M + 1) + 4M + 2$</td>
</tr>
</tbody>
</table>
Complexity of JIDF Algorithms

![Graph showing complexity of JIDF algorithms]
Applications : Interference suppression for CDMA

Parameters : Uplink scenario, QPSK symbols, $K$ users, $N$ chips per symbol and $L$ propagation paths, receiver filter has $M = N + L_p - 1$ taps.
Applications: Interference suppression for CDMA

Parameters: Uplink scenario, QPSK symbols, $K$ users, $N$ chips per symbol and $L$ propagation paths, receiver filter has $M = N + L_p - 1$ taps.

![Graph showing BER performance for different methods across varying $E_b/N_0$ and $K$.]
Applications: STAP for Radar Systems

- The system under consideration is a pulsed Doppler radar residing on an airborne platform.
- The radar antenna is a uniformly spaced linear antenna array consisting of \( N \) elements. The radar returns are collected in a coherent processing interval (CPI).
- The \( M \times 1 \) radar space-time snapshot \( r(i) \) is then expressed for each of the two hypotheses in the following form
  \[
  H_0: r(i) = v(i); \\
  H_1: r(i) = as + v(i);
  \]
- **Problem**: to design a spatial-temporal beamformer with limited training.
Applications: STAP for Radar Systems

Airborne radar system parameters:

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Antenna array</td>
<td>Sideway-looking array (SLA)</td>
</tr>
<tr>
<td>Carrier frequency ($f_c$)</td>
<td>450 MHz</td>
</tr>
<tr>
<td>Transmit pattern</td>
<td>Uniform</td>
</tr>
<tr>
<td>PRF ($f_r$)</td>
<td>300 Hz</td>
</tr>
<tr>
<td>Platform velocity ($v$)</td>
<td>75 m/s</td>
</tr>
<tr>
<td>Platform height ($h$)</td>
<td>9000 m</td>
</tr>
<tr>
<td>Clutter-to-Noise ratio (CNR)</td>
<td>40 dB</td>
</tr>
<tr>
<td>Elements of sensors ($N$)</td>
<td>8</td>
</tr>
<tr>
<td>Number of Pulses ($J$)</td>
<td>8</td>
</tr>
</tbody>
</table>
Applications: STAP for Radar Systems
Applications: STAP for Radar Systems

![Graph showing SINR vs. Target Doppler Frequency](image)

- Optimum MVDR
- SMI MVDR
- LR-EIG
- LR-Krylov
- LR-JIO
- LR-JIDF
- SA-MVDR
- KA-MVDR
Applications: UWB communications

- We apply the a variation of the JIDF scheme called SAABF to the downlink of a multiuser BPSK DS-UWB system and evaluate their performance against existing methods.
- In all numerical simulations, the pulse shape adopted is the RRC pulse with the pulse-width 0.375ns.
- The spreading codes are generated randomly with a spreading gain of 24 and the data rate of the communication is approximately 110Mbps.
- The standard IEEE 802.15.4a channel model for the NLOS indoor environment is employed.
- We assume that the channel is constant during the whole transmission.
- The sampling rate at the receiver is assumed to be 8GHz that is the same as the standard channel model and the observation window length $M$ for each data symbol is set to 120 samples.
Applications : UWB communications

Parameters : BER performance of different algorithms for a SNR=16dB and 3 users. The following parameters were used : full-rank LMS ($\mu = 0.075$), full-rank RLS ($\lambda = 0.998$, $\delta = 10$), MSWF-LMS ($D = 6$, $\mu = 0.075$), MSWF-RLS ($D = 6$, $\lambda = 0.998$), AVF ($D = 6$), SAABF (1,3,M)-LMS ($\mu_w = 0.1$, $\mu_\psi = 0.2$, 2 iterations) and SAABF (1,3,M)-RLS ($\lambda = 0.998$, $\delta = 0.1$, 1 iteration).
Applications: UWB communications

Parameters: BER performance of the proposed SAABF scheme versus the number of training symbols for a SNR=16dB. The number of users is 3 and the following parameters were used: SAABF-RLS ($\lambda = 0.98$, $\delta = 10$).
Applications: UWB communications

SNR=12dB

BER

Full-rank LMS
Full-rank RLS
MSWF-RLS
AVF
SAABF-RLS
MMSE

7 Users

BER

SNR

Full-rank LMS
Full-rank RLS
MSWF-RLS
AVF
SAABF-RLS
MMSE
Model-order selection techniques

- Basic principle: to determine the best fit between observed data and the model used.
- General approaches to model-order selection:
  - Setting of upper bounds on models with "some" prior knowledge: one of the most used in communications.
  - Akaike's information theoretic criterion: works well, requires a large number of computations, not suitable to time-varying scenarios.
  - Minimum description length (MDL): also works well, not suitable to time-varying scenarios.
  - Adaptive filtering approach: use for adaptive algorithms with dynamic lengths, work well and have lower complexity than prior art.
Akaike Information Criterion

– Basic principle: employs information entropy to perform model order selection.
– Method:

\[ AIC = 2k - 2l(\hat{\theta}), \]

where
\( k \) is the number of parameters
\( \hat{\theta} \) is the ML estimate
\( l(\cdot) \) is the log likelihood function.

– The lower the AIC the better the model selected.
– It is more suitable to ML estimation problems.
Minimum Description Length

- Basic principle: given a data set and competing statistical models, the best one is that which provides the shortest description length.
- Method:

\[
MDL = \frac{1}{2} m \ln N - l(\hat{\theta}),
\]

where
\(N\) is the number of samples
\(m\) is the number of independently adjusted parameters
\(l(\cdot)\) is the log likelihood function.

- The shortest the MDL the better the model selected.
- It is more suitable to ML estimation problems.
- The MDL converges to the true model order.
Model-Order Selection for Time-Varying Scenarios

– Approaches used for reduced-rank techniques:
  – Testing of orthogonality conditions between columns of transformation matrix $S_D(i)$ [12]:
    – used with the MSWF for selecting the rank $D$.
  – Cross-validation of data [24]:
    – used with the AVF,
    – works well but can be complex since the algorithms sometimes selects $D$ quite large.
    – This can be a problem if $M$ is large and $D$ approaches it.
  – Use of a priori values of least-squares type cost functions with lower and upper bounds:
    – works very well and it is simple to use and design [12, 17, 35].
    – It can be easily extended when the designer has multiple parameters with orders to adjust.
Model-order selection with the JIO-MVDR algorithm

– Consider the exponentially weighted *a posteriori* least-squares type cost function described by

\[ C(S_D(i-1), \tilde{w}^{(D)}(i-1)) = \sum_{l=1}^{i} \alpha^{i-l} |\tilde{w}^H, (D)(i-1)S_D(i-1)r(l)|^2, \]

where \( \alpha \) is the forgetting factor and \( \tilde{w}^{(D)}(i-1) \) is the reduced-rank filter with rank \( D \).

– For each time interval \( i \), we can select the rank \( D_{\text{opt}} \) which minimizes \( C(S_D(i-1), \tilde{w}^{(D)}(i-1)) \) and the exponential weighting factor \( \alpha \) is required as the optimal rank varies as a function of the data record.

– The key quantities to be updated are \( S_D(i), \tilde{w}(i), \bar{a}(\theta_k) \) and \( \bar{P}(i) \) (RLS algorithm).
Model-order selection with JIO-MVDR algorithms

- Let us define the following extended matrix $S^{(D)}(i)$ and the extended reduced-rank beamformer $\bar{w}^{(D)}(i)$ as follows:

\[
S^{(D)}(i) = \begin{bmatrix}
s_{1,1} & \cdots & s_{1,D_{\text{min}}} & \cdots & s_{1,D_{\text{max}}} \\
\vdots & \ddots & \vdots & \ddots & \vdots \\
s_{M,1} & \cdots & s_{M,D_{\text{min}}} & \cdots & s_{M,D_{\text{max}}}
\end{bmatrix}
\]

and

\[
\bar{w}^{(D)}(i) = \begin{bmatrix}
w_1 \\
\vdots \\
w_{D_{\text{min}}} \\
\vdots \\
w_{D_{\text{max}}}
\end{bmatrix}
\]

- $S^{(D)}(i)$ and $\bar{w}^{(D)}(i)$ are updated along with the associated quantities $\bar{a}(\theta_k)$ and $\bar{P}(i)$ for the maximum allowed rank $D_{\text{max}}$. 
Model-order selection with JIO-MVDR algorithms

- The model-order selection algorithm determines the rank that is best for each time instant $i$ using the cost function.

- The model-order selection algorithm is then given by

$$D_{\text{opt}} = \arg \min_{D_{\text{min}} \leq d \leq D_{\text{max}}} \mathcal{C}(S_{D}(i - 1), \bar{w}^{(D)}(i - 1))$$

where

- $d$ is an integer,
- $D_{\text{min}}$ and $D_{\text{max}}$ are the minimum and maximum ranks allowed for the reduced-rank filter, respectively.
Model-order selection with JIO-MVDR algorithms

SINR performance of LCMV (a) SG and (b) RLS algorithms against snapshots with $M = 24$, $SNR = 12$ dB with automatic rank selection.
Model-order selection with JIDF algorithms

– Consider the following exponentially weighed \textit{a posteriori} least-squares type cost function

\[ C(\tilde{w}^{(D)}, v^{(N_I)}, D) = \sum_{l=1}^{i} \alpha^{i-l} \left| d(l) - \tilde{w}^H(l)D(l)R_o(l)v^*, (N_I)(l) \right|^2 \]

where
\( \alpha \) is the forgetting factor,
\( \tilde{w}^{(D)}(i - 1) \) is the reduced-rank filter with rank \( D \) and
\( v^{(N_I)}(i) \) is the interpolator filter with rank \( N_I \).

– For each time interval \( i \) and a given decimation pattern and \( B \), we can select \( D \) and \( N_I \) which minimizes \( C(\tilde{w}^{(D)}, v^{(N_I)}, D) \).
Model-order selection with JIDF algorithms

- The model-order selection algorithm that chooses the best lengths $D_{\text{opt}}$ and $N_{I_{\text{opt}}}$ for the filters $v(i)$ and $\bar{w}(i)$, respectively, is given by

$$\{D_{\text{opt}}, N_{I_{\text{opt}}}\} = \arg \min_{N_{I_{\text{min}}} \leq n \leq N_{I_{\text{max}}}} \{\min_{D_{\text{min}} \leq d \leq D_{\text{max}}} C(\bar{w}(d), v(n), D)\}$$

where

d and $n$ are integers,

$D_{\text{min}}$ and $D_{\text{max}}$ and

$N_{I_{\text{min}}}$ and $N_{I_{\text{max}}}$ are the minimum and maximum ranks allowed for $\bar{w}(i)$ and $v(i)$, respectively.
Model-order selection with JIDF algorithm

SINR performance against rank (D) for the analyzed schemes using LMS and RLS algorithms.
Applications, perspectives and future work

– **Applications**: interference suppression, beamforming, channel estimation, echo cancellation, target tracking, wireless sensor networks, signal compression, radar, control, seismology, etc.

– **Perspectives**:
  - Work in this field is not fully explored.
  - Many unsolved problems when dimensions become large: estimation, tracking, general acquisition, networks, distributed problems.

– **Future work**:
  - Information theoretic study of very large observation data: performance limits as $M$ goes to infinity.
  - Development of vector and matrix-based parameter estimates as opposed to current scalar parameter estimation of existing methods.
  - Distributed reduced-rank processing.
Concluding remarks

– Reduced-rank signal processing is a powerful set of tools that allow the processing of large data vectors, enabling a substantial reduction in training and complexity.

– An overview of reduced-rank techniques, detailing eigen-decomposition methods and the MSWF, was presented along with applications in communications and sensor array systems.

– A family of reduced-rank algorithms based on JIO techniques was presented along with applications.

– A recently proposed reduced-rank scheme called JIDF was also briefly reviewed with applications.

– Several applications have been considered as well as a number of future investigation topics have been discussed.
Questions?
Thank you!

Contact:
Dr. R. C. de Lamare
Communications Research Group
University of York
Website: http://www-users.york.ac.uk/~rcdl500/
E-mail: rcdl500@ohm.york.ac.uk
References


