

A SINR Maximizing Interpolation-and-Decimation-based Dimensionality Reduction Technique, with Application to Beamforming

Aline de Oliveira, José Mauro Fortes, Fabian David Backx, Raimundo Sampaio-Neto

Abstract—We present a dimensionality reduction technique based on a joint interpolation and decimation scheme, with application to beamforming. The dimensionality reduction is achieved by a two step procedure: interpolation followed by decimation. The array snapshots are interpolated by a finite impulse response (FIR) filter in order to generate correlation between its samples. The decimation stage then discards some samples from the correlated interpolator output signal, effectively reducing the snapshots' length. A notable point of this technique is the elegant and effective way to design the interpolation filter. The design is such that, for a given decimation pattern, the interpolation filter maximizes the signal-to-interference-and-noise ratio (SINR) at the output of the decimation stage. The optimization of the reduced dimensionality stage is made independently of the final application filtering stage, allowing the proposed scheme to be combined with any interference-suppressive or detection filter of choice. Investigation of this technique in light of the particularities of the beamforming signal model led to, here proposed, simplifications that allowed for a significant reduction of its overall complexity. Comparison with renowned robust rank reduction techniques show that the proposed approach has an excellent SINR loss figure of merit performance with superior robustness and low computational complexity.

I. INTRODUCTION

Interpolation and decimation algorithms were extensively studied for sampling rate alteration and related applications [1]. In these algorithms, the input signal is filtered prior to the decimation stage in order to avoid aliasing. As the sampling rate is an important cost factor in digital signal processors implementations, so is the length of the input data in ever increasing sophisticated algorithms.

One of this paper's authors investigated the interpolation and decimation concept focusing on the dimensionality reduction of the observed data prior to the digital processing algorithms. The idea was to design the interpolation filter aiming at the minimization of a certain cost function. At first, this idea originated an algorithm where the dimensionality-reduction stage is coupled with the final application filter (the detection filter for example), adaptively adjusting both the interpolation and detection filter weights [2]. This idea proved to give

Aline de Oliveira and Fabian David Backx are with Instituto de Pesquisas da Marinha, Rio de Janeiro, RJ, 21931-095, Brazil, e-mail: {aline.oliveira, fabian}@marinha.mil.br.

José Mauro Fortes and Raimundo Sampaio-Neto are with Pontifícia Universidade Católica do Rio de Janeiro, Rio de Janeiro, RJ, 22451-900, Brazil, e-mail: {jmfortes, raimundo}@cetuc.puc-rio.br.

Digital Object Identifier: 10.14209/jcis.2020.1

excellent results at low computational cost in Direct Sequence Code Division Multiple Access (DS-CDMA) communication scenarios using Minimum Mean Square Error (MMSE) filters [3], [4], [5].

Building on this concept, another approach was then proposed: to design an interpolator filter that maximizes the signal-to-noise ratio (SNR) at the output of the decimation stage, irrespective of the final application filter. Indeed, differently from [2], the optimization of the reduced dimensionality stage is made independently of the filtering stage, allowing the proposed scheme to be combined with any interference-suppressive or detection filter of choice. Since the interpolation filter is now decoupled from the application filter, the advantage of this stand-alone dimensionality reduction block relies on the fact that it can be deployed upstream from any desired application filter, e.g. any kind of detection or estimation filter, without the need of redesigning it from the application cost function, as it has to be done for different applications [2], [3], [4], [5]. This alternative approach was tested in Ultra Wide Band (UWB) communication scenarios, and had excellent results [6], [7], [8].

The aforementioned method was further improved, leading to the design of a decoupled interpolation filter that maximizes the signal-to-interference-and-noise ratio (SINR), instead of only the SNR as in [6], [7], [8]. The design of this new interpolator is creative and effective. This method was applied for DS-CDMA and UWB communication scenarios and reported in Portuguese in [9] and [10]. This method will be now on called: joint interpolation and decimation scheme (JIDS).

In this paper, we recast the JIDS algorithm for beamforming applications and specialize it considering the particularities of this specific problem. We propose simplified methods for selecting the algorithm parameters and for defining the best decimation strategy: we investigate the dependence of two of the JIDS parameters, the decimation factor, F , and the interpolation filter length, L_v , and obtain a straightforward method to set the length of the interpolation filter. We also derive a new low complexity criterium for selecting the best decimation pattern. The proposed specialized JIDS for beamforming will be referred to as JIDSB.

The Minimum Variance Distortionless Response (MVDR) beamforming filter [11] is a well-known beamforming technique that exploits the second-order statistics of the interference vector to minimize the array variance while constraining

the array response towards the direction of the signal of interest (SOI). The Minimum Power Distortionless Response (MPDR) beamforming filter, as denoted in [11], exploits the second-order statistics of the received vector to minimize the array output power while constraining the array response towards the direction of the SOI. When the direction of arrival (DOA) of the SOI is known exactly the output of the MPDR resumes to the output of the MVDR [11].

There are many techniques available to implement the MVDR filter with less computational burden, basically to avoid the MVDR matrix inversion step, e.g. the stochastic-gradient (SG) [12], [13] technique. However, full rank algorithms usually require a large number of snapshots to reach the steady-state. In large antenna array arrangements, this may cause degradation in convergence speed, especially in environments where a small support of independent and identically distributed (IID) samples are available for estimation of the statistical quantities [14].

Reduced-rank techniques can mitigate the effects of these drawbacks. Principal Components (PC) [15], [16] and Cross-Spectral Metric (CSM) [17] are examples of reduced-rank filtering schemes based on an eigen-decomposition. The Multistage Wiener Filter (MWF) achieves rank reduction through the Krylov subspace, which has the added benefit of a further reduction in computational complexity based for example on the Lanczos [18], [19], [20] or the Arnoldi [21], [20] algorithms, or a conjugate gradient-based (CG) implementation [22], [23] (CG-MWF). Other rank-reduction algorithms are, for example, the Auxiliary Vector Filter (AVF) [24], [25] that generates a sequence of linear auxiliary filters that converge to the MVDR filter and the family of adaptive joint iterative optimization (JIO) algorithms [26], [27].

Another concern in beamforming is robustness, which indicates how the algorithms perform under certain unfavorable situations, e.g. calibration errors, look of direction errors, distortions caused by source spreading and poor estimation of statistical quantities due to small sample support sizes. This problem may be especially dramatic when the MPDR beamformer filter is used. Many approaches have been proposed to improve the beamformers robustness, for example, adding extra quadratic (diagonal loading), linear point or derivative constraints, [28], [12], [29], [30] and robust estimation using random matrices theory [31] or eigenspace based robust beamformers [32]. It is within this context that the JIDSB algorithm is inserted.

We apply the MVDR detection filter to the output of the dimensionality reduction stage and compare the JIDSB results in terms of SINR loss with the full rank MVDR filter and several rank reduction techniques such as PC, CSM and CG-MWF. In terms of robustness, we compare the JIDSB with the former methods with diagonal loading when the MPDR filter is applied.

This paper is organized as follows. Section II describes the signal model. Section III combines the dimensionality reduction transforms with the beamformer weighting vector. Section IV recasts the JIDS, specialized here for the beamforming application. The proposed simplification procedures are explained in subsection IV-B and IV-C. Subsection IV-D

addresses the computational complexity issue. The performance assessment examples of the proposed specialized and simplified JIDS (i.e. JIDSB) are provided in Section V. Finally, conclusions are given in Section VI.

II. SYSTEM MODEL

We consider a beamforming application with a uniform linear array (ULA) with M elements. The sensor array received vector at the i -th time snapshot $\mathbf{r}(i) \in \mathbb{C}^{M \times 1}$ is given by

$$\mathbf{r}(i) = \mathbf{s}(\theta_0)b_0(i) + \underbrace{\mathbf{i}(i) + \mathbf{n}(i)}_{\mathbf{x}(i)}, \quad (1)$$

where without loss of generality, the signal of interest is represented as b_0 , a complex random variable with power $E[|b_0|^2] = \sigma_0^2$ and steering vector $\mathbf{s}(\theta_0) = [1, e^{-j\frac{2\pi d}{\lambda_c} \sin(\theta_0)}, \dots, e^{-j\frac{2\pi(M-1)d}{\lambda_c} \sin(\theta_0)}]^T$, where λ_c is the carrier wavelength and d is the inter-element spacing of the ULA. The term $\mathbf{x}(i) \in \mathbb{C}^{M \times 1}$ is the interference plus noise vector, where

$$\mathbf{i}(i) = \mathbf{A}(\Theta)\mathbf{b}(i), \quad (2)$$

$\Theta = [\theta_1, \dots, \theta_Q]^T \in \mathbb{R}^{Q \times 1}$ is the vector of direction of arrivals of Q ($Q < M$) narrowband interference signals impinging on the ULA, $\mathbf{A}(\Theta) = [\mathbf{s}(\theta_1), \dots, \mathbf{s}(\theta_Q)] \in \mathbb{C}^{M \times Q}$ is the complex matrix composed of the interference steering vectors, $\mathbf{s}(\theta_k) \in \mathbb{C}^{M \times 1}$, $k = 1, \dots, Q$, given by:

$$\mathbf{s}(\theta_k) = [1, e^{-j\frac{2\pi d}{\lambda_c} \sin(\theta_k)}, \dots, e^{-j\frac{2\pi(M-1)d}{\lambda_c} \sin(\theta_k)}]^T. \quad (3)$$

The elements of the vector $\mathbf{b}(i) = [b_1(i), \dots, b_Q(i)]^T \in \mathbb{C}^{Q \times 1}$ are modeled as random variables from uncorrelated zero-mean, circular complex processes, with variances given by $\sigma_1^2, \sigma_2^2, \dots, \sigma_Q^2$. The vector $\mathbf{n}(i) \in \mathbb{C}^{M \times 1}$ is the complex vector of sensor noise, which is assumed to be a zero-mean spatially and temporally circular complex Gaussian vector. The beamformer output is then given by

$$z(i) = \mathbf{w}^H \mathbf{r}(i), \quad (4)$$

where $\mathbf{w} = [w_1, \dots, w_M]^T \in \mathbb{C}^{M \times 1}$ is the complex weighting vector.

The beamformer weighting vector \mathbf{w} can be designed to maximize the SINR in $z(i)$, according to the MVDR criterion [11]

$$\min_{\mathbf{w}} \mathbf{w}^H \mathbf{R} \mathbf{w}, \text{ subject to } \mathbf{w}^H \mathbf{s}(\theta_0) = 1, \quad (5)$$

where $\mathbf{R} = \mathbb{E}[\mathbf{x}(i)\mathbf{x}^H(i)]$ is the autocorrelation matrix of the interference and noise. The well-known solution of (5) is the optimal weighting vector given by:

$$\mathbf{w}_{\text{op}} = \frac{\mathbf{R}^{-1} \mathbf{s}(\theta_0)}{\mathbf{s}^H(\theta_0) \mathbf{R}^{-1} \mathbf{s}(\theta_0)}. \quad (6)$$

It is easy to show that the SINR achieved with the optimal filter of (6) is given by

$$\text{SINR}(\mathbf{w}_{\text{op}}) = \sigma_0^2 \mathbf{s}^H(\theta_0) \mathbf{R}^{-1} \mathbf{s}(\theta_0). \quad (7)$$

When the autocorrelation matrix is not known *a priori*, it has to be estimated from the observed data. In statistical stationary

signal scenarios, the autocorrelation matrix can be estimated from the available sample support with N_s snapshots as

$$\hat{\mathbf{R}} = \frac{1}{N_s} \sum_{i=1}^{N_s} \mathbf{x}(i)\mathbf{x}^H(i). \quad (8)$$

The solution in (6) that entails the computation of the inverse of the estimated autocorrelation matrix, $\hat{\mathbf{R}}$, is called the MVDR sample matrix inversion (SMI) beamformer.

The optimization problem that arises when we use the autocorrelation matrix of the whole incoming signal, $\mathbf{R} = \mathbb{E}[\mathbf{r}(i)\mathbf{r}^H(i)]$, is known as MPDR [11]. The MPDR and the MVDR solutions are identical in conditions of perfect knowledge of the autocorrelation matrix and the desired steering vector. But SMI-based MPDR beamformers are known to suffer from performance degradation [11], [28], [33]. The performance degradation is due to signal cancellation, termed as signal self-nulling. This problem becomes especially dramatic in practical scenarios, when there are mismatches between the assumed array response and the true array response. This situation arises, for example, when there is a finite sample support for estimating the autocorrelation matrix.

Diagonal loading is a popular approach to improve the MPDR-SMI beamformer robustness. It is derived by imposing an additional quadratic constraint either on the Euclidian norm of the weight vector itself or on its difference from a desired weight vector [28]. The estimated autocorrelation matrix added with a diagonal loading $\gamma \in \mathbb{R}^+$, $\hat{\mathbf{R}}_{DL}$ is given by

$$\hat{\mathbf{R}}_{DL} = \hat{\mathbf{R}} + \gamma \mathbf{I}. \quad (9)$$

III. DIMENSIONALITY REDUCTION TECHNIQUE FOR BEAMFORMING APPLICATION

Consider a dimensionality reduction transformation matrix $\mathbf{T} \in \mathbb{C}^{D \times M}$. The observed i -th snapshot $\mathbf{r}_D(i) \in \mathbb{C}^{D \times 1}$ after the dimensionality reduction, given by

$$\mathbf{r}_D(i) = \mathbf{T}\mathbf{r}(i), \quad (10)$$

is then processed by the beamforming filter, $\mathbf{w}_D \in \mathbb{C}^{D \times 1}$, whose output $z_D(i)$ is given by

$$z_D(i) = \mathbf{w}_D^H \mathbf{r}_D(i). \quad (11)$$

The complex weighting vector $\mathbf{w}_D = [w_1, \dots, w_D]^T \in \mathbb{C}^{D \times 1}$ is designed according to the MVDR criteria for the reduced observation \mathbf{r}_D and is given by

$$\mathbf{w}_D = \frac{\mathbf{R}_D^{-1} \mathbf{s}_D(\theta_0)}{\mathbf{s}_D^H(\theta_0) \mathbf{R}_D^{-1} \mathbf{s}_D(\theta_0)}, \quad (12)$$

where $\mathbf{R}_D = \mathbf{T}\mathbf{T}^H = \mathbb{E}[\mathbf{x}_D(i)\mathbf{x}_D^H(i)]$ is the autocorrelation matrix of the interference plus noise after the dimensionality reduction stage, where $\mathbf{R} \in \mathbb{C}^{M \times M}$ is the autocorrelation matrix of the interference plus noise of the original data and $\mathbf{s}_D(\theta_0) = \mathbf{T}\mathbf{s}(\theta_0)$ is the desired signal steering vector after the dimensionality reduction stage. The block diagram of this process is illustrated in Fig. 1.

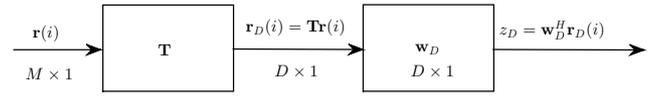


Fig. 1. Block diagram of dimensionality reduction stage followed by MVDR beamformer filter.

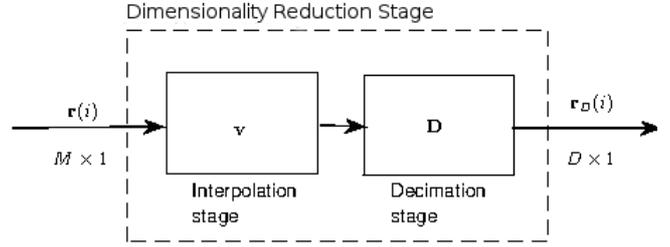


Fig. 2. Illustration of the JIDS dimensionality reduction stage.

IV. JIDS DIMENSIONALITY REDUCTION TECHNIQUE

The JIDS dimensionality reduction technique was presented by one of the authors in a regional conference in Portuguese [9] for DS-CDMA and UWB communications. For the sake of completeness, this section gives an overview of the method modifying it to take advantage of the beamforming system model. In general terms, the JIDS is based on two operations: interpolation and decimation, as depicted in Fig 2. At the interpolation stage the i -th received snapshot $\mathbf{r}(i)$ is filtered by $\mathbf{v} \in \mathbb{C}^{L_v \times 1}$ ($L_v \ll M$) in order to correlate its components before the decimation stage. The decimation stage is implemented by means of a decimation matrix, \mathbf{D} , that selects certain components, reducing the original dimension M by a factor of F . The resulting vector length is $D = \lfloor M/F \rfloor$, where $\lfloor x \rfloor$ is the operation that selects the largest integer not greater than x . For a uniform decimation by a factor of F , there are in fact F possible patterns, $l \in \{0, \dots, F-1\}$. The index l , that designates the decimation pattern, $\mathbf{D}_l \in \mathbb{C}^{D \times M}$, corresponds to the row of the first component of $\mathbf{r}(i)$ selected by the decimation block, that is

$$\mathbf{D}_l = \begin{pmatrix} \phi_{l,0} \\ \phi_{l,1} \\ \vdots \\ \phi_{l,D-1} \end{pmatrix}, \quad (13)$$

where

$$\phi_{l,i} = \underbrace{[0, \dots, 0, 1, 0, \dots, 0]}_{iF+l}. \quad (14)$$

The JIDS dimensionality reduction technique makes a joint choice of the interpolation filter, \mathbf{v}_l , and the decimation pattern, \mathbf{D}_l as will be explained in the following subsection.

A. An Effective Design of the Interpolation Filter Specialized for Beamforming

Given a decimation pattern l , we seek the interpolation filter \mathbf{v}_l^* that maximizes the SINR at the output of the dimensionality reduction stage. Furthermore, we also seek the decimation pattern l^* that results in the highest SINR among

all F possible decimation patterns, $l \in \{0, \dots, F-1\}$. The problem of choosing the decimation pattern, l^* , will be shown to simplify to a trivial comparison of scalars.

In the following, we address the problem of finding the interpolation filter \mathbf{v}_l^* that maximizes the SINR at the output of the dimensionality reduction stage, given a decimation pattern l . The output of the dimensionality reduction stage using the l -th decimation pattern, at the i -th snapshot, is given by

$$\mathbf{r}_{D_l}(i) = \mathbf{D}_l \mathcal{V}_l \mathbf{r}(i) \quad (15)$$

$$= \mathbf{D}_l \mathcal{V}_l (\mathbf{s}(\theta_0) b_0(i) + \mathbf{i}(i) + \mathbf{n}(i)) \quad (16)$$

$$= \mathbf{s}_{D_l} b_0(i) + \mathbf{i}_{D_l}(i) + \mathbf{n}_{D_l}(i), \quad (17)$$

where $\mathcal{V}_l \in \mathbb{C}^{M \times M}$ is a Toeplitz matrix that implements the discrete convolution between $\mathbf{v}_l \in \mathbb{C}^{L_v \times 1}$ and $\mathbf{r}(i) \in \mathbb{C}^{M \times 1}$. The first column of \mathcal{V}_l is given by $[\mathbf{v}_l^T, 0, \dots, 0]^T \in \mathbb{C}^{M \times 1}$. Due to the convolution commutation property,

$$\mathcal{V}_l \mathbf{r}(i) = \mathcal{R}(i) \mathbf{v}_l, \quad (18)$$

where $\mathcal{R} \in \mathbb{C}^{M \times L_v}$ is a Toeplitz matrix whose first column is given by $\mathbf{r}(i) \in \mathbb{C}^{M \times 1}$. Using (18) we can rewrite (15) as

$$\mathbf{r}_{D_l}(i) = \mathbf{D}_l \mathcal{R}(i) \mathbf{v}_l. \quad (19)$$

Similarly, \mathbf{s}_{D_l} , $\mathbf{i}_{D_l}(i)$ and $\mathbf{n}_{D_l}(i)$ are defined as

$$\mathbf{s}_{D_l} = \mathbf{D}_l \mathcal{V}_l \mathbf{s}(\theta_0) = \mathbf{D}_l \mathcal{S} \mathbf{v}_l, \quad (20)$$

$$\mathbf{i}_{D_l}(i) = \mathbf{D}_l \mathcal{V}_l \mathbf{i}(i) = \mathbf{D}_l \mathcal{I}(i) \mathbf{v}_l, \quad (21)$$

$$\mathbf{n}_{D_l}(i) = \mathbf{D}_l \mathcal{V}_l \mathbf{n}(i) = \mathbf{D}_l \mathcal{N}(i) \mathbf{v}_l, \quad (22)$$

where $\mathcal{S} \in \mathbb{C}^{M \times L_v}$, $\mathcal{I} \in \mathbb{C}^{M \times L_v}$ and $\mathcal{N} \in \mathbb{C}^{M \times L_v}$ are Toeplitz matrices with their first columns given respectively by $\mathbf{s}(\theta_0) \in \mathbb{C}^{M \times 1}$, $\mathbf{i}(i) \in \mathbb{C}^{M \times 1}$ and $\mathbf{n}(i) \in \mathbb{C}^{M \times 1}$.

The SINR after the dimensionality reduction stage using the l -th decimation pattern is given by (dropping the snapshot index i for convenience)

$$\text{SINR}_l = \frac{\sigma_0^2 \|\mathbf{s}_{D_l}\|^2}{E[\|\mathbf{i}_{D_l} + \mathbf{n}_{D_l}\|^2]}. \quad (23)$$

The filter \mathbf{v}_l^* that maximizes (23) is the one that satisfies

$$\mathbf{v}_l^* = \arg \max_{\mathbf{v}} \frac{\|\mathbf{s}_{D_l}\|^2}{E[\|\mathbf{i}_{D_l} + \mathbf{n}_{D_l}\|^2]}, \quad (24)$$

which is equivalent to

$$\mathbf{v}_l^* = \arg \max_{\mathbf{v}} \frac{\|\mathbf{s}_{D_l}\|^2}{E[\|\mathbf{r}_{D_l}\|^2]}, \quad (25)$$

since

$$\begin{aligned} \frac{E[\|\mathbf{r}_{D_l}\|^2]}{\sigma_0^2 E[\|\mathbf{s}_{D_l}\|^2]} &= \frac{E[\|\mathbf{s}_{D_l} b_0 + \mathbf{i}_{D_l} + \mathbf{n}_{D_l}\|^2]}{\sigma_0^2 \|\mathbf{s}_{D_l}\|^2} \\ &= \frac{\sigma_0^2 \|\mathbf{s}_{D_l}\|^2 + E[\|\mathbf{i}_{D_l} + \mathbf{n}_{D_l}\|^2]}{\sigma_0^2 \|\mathbf{s}_{D_l}\|^2} \\ &= 1 + \frac{1}{\text{SINR}_l}. \end{aligned} \quad (26)$$

The numerator in (25) may be written as

$$\|\mathbf{s}_{D_l}\|^2 = \|\mathbf{D}_l \mathcal{S} \mathbf{v}_l\|^2 \quad (27)$$

$$= \mathbf{v}_l^H \mathbf{A}_l \mathbf{v}_l, \quad (28)$$

where $\mathbf{A}_l \in \mathbb{C}^{L_v \times L_v}$ is the symmetric, hermitian, non-negative matrix given by

$$\mathbf{A}_l = \mathcal{S}^H \mathbf{D}_l^H \mathbf{D}_l \mathcal{S} \quad (29)$$

$$= \mathcal{S}^H \text{diag}(\mathbf{p}_l) \mathcal{S}. \quad (30)$$

The matrix $\text{diag}(\mathbf{p}_l)$ is a diagonal matrix that is formed with the elements of the vector \mathbf{p}_l along the main diagonal. The vector \mathbf{p}_l identifies the l -th, $l \in \{0, \dots, F-1\}$, decimation pattern: it has zeros at the positions where the elements will be discarded and ones where the elements will be selected, that is, the elements with indices $\{l, l+F, l+2F, \dots, l + \lfloor M/F \rfloor\}$ are retained.

The denominator in (25) may be written as

$$E[\|\mathbf{r}_{D_l}\|^2] = E[\|\mathbf{D}_l \mathcal{R} \mathbf{v}_l\|^2] \quad (31)$$

$$= \mathbf{v}_l^H \mathbf{B}_l \mathbf{v}_l, \quad (32)$$

where $\mathbf{B}_l \in \mathbb{C}^{L_v \times L_v}$ is a hermitian symmetric, non-negative matrix given by

$$\mathbf{B}_l = E[\mathcal{R}^H \mathbf{D}_l^H \mathbf{D}_l \mathcal{R}] \quad (33)$$

$$= E[\mathcal{R}^H \text{diag}(\mathbf{p}_l) \mathcal{R}]. \quad (34)$$

The maximization problem in (25) can be restated as

$$\mathbf{v}_l^* = \arg \max_{\mathbf{v}} f_l(\mathbf{v}), \quad (35)$$

where $f_l(\mathbf{v})$ is defined as

$$f_l(\mathbf{v}) = \frac{\mathbf{v}^H \mathbf{A}_l \mathbf{v}}{\mathbf{v}^H \mathbf{B}_l \mathbf{v}} = \mathbf{v}^H \mathbf{A}_l \mathbf{v} (\mathbf{v}^H \mathbf{B}_l \mathbf{v})^{-1}. \quad (36)$$

The gradient of $f_l(\mathbf{v})$ is computed as

$$\nabla_{\mathbf{v}} f_l(\mathbf{v}) = -\mathbf{v}^H \mathbf{A}_l \mathbf{v} (\mathbf{v}^H \mathbf{B}_l \mathbf{v})^{-2} \mathbf{B}_l \mathbf{v} + (\mathbf{v}^H \mathbf{B}_l \mathbf{v})^{-1} \mathbf{A}_l \mathbf{v}. \quad (37)$$

The values that null (37), or equivalently,

$$-(\mathbf{v}^H \mathbf{A}_l \mathbf{v}) \mathbf{B}_l \mathbf{v} + (\mathbf{v}^H \mathbf{B}_l \mathbf{v}) \mathbf{A}_l \mathbf{v} = 0, \quad (38)$$

must satisfy

$$\mathbf{A}_l \mathbf{v} = \lambda \mathbf{B}_l \mathbf{v}, \quad (39)$$

or

$$\mathbf{F}_l \mathbf{v} = \lambda \mathbf{v}, \quad (40)$$

where $\mathbf{F}_l = \mathbf{B}_l^{-1} \mathbf{A}_l$ and λ is the scalar given by

$$\lambda = \frac{\mathbf{v}^H \mathbf{A}_l \mathbf{v}}{\mathbf{v}^H \mathbf{B}_l \mathbf{v}} = f_l(\mathbf{v}). \quad (41)$$

We notice that (40) is the eigenvalue equation of matrix \mathbf{F}_l . Therefore, vector \mathbf{v} that solves (40) must be the eigenvector of \mathbf{F}_l associated to λ . But as can be seen comparing (41) with (36), λ is the SINR itself. Thus, in order to maximize the SINR we need to find the eigenvector associated to the largest eigenvalue of \mathbf{F}_l . By doing so, we are choosing the interpolation filter, \mathbf{v}_l^* , that will produce the maximal SINR for the l -th decimation pattern. In summary, the interpolation filter \mathbf{v}_l^* that maximizes the SINR at the output of the dimensionality reduction stage, given a decimation pattern l , is the eigenvector associated to the largest eigenvalue of \mathbf{F}_l .

In the following, we address the problem of finding the decimation pattern l^* that results in the highest SINR among all F decimation patterns. It turns out that the answer is simple: we just have to compare the largest eigenvalue of \mathbf{F}_l for $l \in \{0, \dots, F-1\}$ and select l^* as the decimation pattern that produces \mathbf{F}_{l^*} with the largest eigenvalue. This process can be done in a parallel multiple branch structure with F branches, where each branch uses a different decimation pattern followed by a simple scalar comparison. By doing so, we are choosing the interpolation filter $\mathbf{v}_{l^*}^*$, that will produce the maximal SINR among all F decimation patterns.

We can now recast the JIDS for beamforming applications:

- 1) Construct the Toeplitz matrix of the desired steering vector $\mathbf{s}(\theta_0)$, \mathcal{S} ;
- 2) Compute the $L_v \times L_v$ matrix $\mathbf{A}_l = \mathcal{S}^H \text{diag}(\mathbf{p}_l) \mathcal{S}$, as in (30);
- 3) Estimate the $L_v \times L_v$ matrix \mathbf{B}_l in (34) using N_B snapshots as

$$\hat{\mathbf{B}}_l = \frac{1}{N_B} \sum_{i=1}^{N_B} \mathcal{R}^H(i) \text{diag}(\mathbf{p}_l) \mathcal{R}(i); \quad (42)$$

- 4) Compute the $L_v \times L_v$ matrix $\hat{\mathbf{F}}_l = \hat{\mathbf{B}}_l^{-1} \mathbf{A}_l$;
- 5) Compute the largest eigenvalue, $\lambda_{\max, l}$, of $\hat{\mathbf{F}}_l$;
- 6) Repeat steps 2 to 5 for the F possible decimation patterns and choose the decimation pattern l^* that provides the largest eigenvalue of $\hat{\mathbf{F}}_l$

$$l^* = \arg \max_l \lambda_{\max, l}, \quad (43)$$

$$l \in [0, \dots, F-1]. \quad (44)$$

- 7) For the decimation pattern l^* (selected in the previous step), set the interpolation filter $\mathbf{v}_{l^*}^*$ as the eigenvector associated to the largest eigenvalue of $\hat{\mathbf{F}}_{l^*}$.

After determining \mathbf{D} and \mathcal{V} using the steps described above, the JIDS dimensionality reduction transformation matrix $\mathbf{T}_J \in \mathbb{C}^{D \times M}$ is, thus, given by

$$\mathbf{T}_J = \mathbf{D} \mathcal{V}. \quad (45)$$

B. Selection of the Interpolator Length

Selection of the interpolator length, L_v , and the decimation factor, F , may require an extensive search. In this section we will examine the particularities of the beamforming problem in order to suggest a good choice for those parameters adjustments.

Previous work [6] showed that, for scenarios where the observed data is corrupted only by white noise, the best results occurred for interpolation filter lengths equal to the decimation factor $L_v = F$. This may be explained by the fact that $L_v = F$ is the filter length that combines the largest number of samples while preserving the statistical characteristics of the white noise vector. For this choice of L_v , the time interval between the preserved noise samples is greater than the memory of the interpolator filter and the white noise vector remains white after filtering and decimation operations. Indeed, it is straightforward to show that, due to the structure of \mathcal{V}_l in (22), the k -th component $z_k(i)$ of

the filtered noise vector $\mathbf{z}(i) = \mathcal{V}_l \mathbf{n}(i)$ in (22) depends only on the (at most) L_v components $n_k, n_{k-1}, \dots, n_{k-L_v+1}$ of $\mathbf{n}(i)$, that is $z_k(i) = f(n_k, n_{k-1}, \dots, n_{k-L_v+1})$. The net effect of the matrix \mathbf{D}_l in (22) is to select (keep) one out of every F components of $\mathbf{z}(i)$, resulting in $\mathbf{n}_{D_l}(i)$. In this respect, any two adjacent components of $\mathbf{n}_{D_l}(i)$ are indeed two components of $\mathbf{z}(i)$ spaced F components apart of each other. Thus, in order for any two adjacent components of $\mathbf{n}_{D_l}(i)$ to be uncorrelated, $z_k = f(n_k, n_{k-1}, \dots, n_{k-L_v+1})$ and $z_{k+F} = f(n_{k+F}, n_{k+F-1}, \dots, n_{k+F-L_v+1})$ must be uncorrelated. For this to hold, z_k and z_{k+F} cannot have components of $\mathbf{n}(i)$ in common, or equivalently, $k+F-L_v+1 > k$ that is $F > L_v - 1$, and since F and L_v are integers, $F \geq L_v$. Therefore, $L_v = F$ is the condition that combines the largest number of samples while preserving the statistical characteristics of the white noise vector.

That is a good starting point for investigating the parameters settings in beamforming scenarios as well. It is to be expected that in cases where the jammer-to-noise-ratio (JNR) is very low, using $L_v = F$ is the best choice, as it approaches the white noise only scenario. In scenarios where the JNR is very high it may not be the best setting, but it may still be a good choice. We checked this through extensive computer simulations and verified that, for beamforming applications, setting $L_v = F$ is indeed a good setting. In this subsection we will show only two representative results for illustration purposes.

We simulated a beamforming scenario consisting of: $M = 64$ elements; SOI at 0° and SNR = 10 dB. We varied the number of jammers and their JNR and evaluated the SINR loss for the decimation factors of 2, 4, 8 and 16 for a range of filter lengths. The SINR loss (L_{SINR}) was computed as

$$L_{\text{SINR}} = \frac{\|\mathbf{w}_D^H \mathbf{s}_D\|^2}{(\mathbf{w}_D^H \mathbf{R}_D \mathbf{w}_D)(\mathbf{s}^H(\theta_0) \mathbf{R}^{-1} \mathbf{s}(\theta_0))}, \quad (46)$$

where \mathbf{R} is the true autocorrelation of the noise and interference known *a priori* and

$$\mathbf{R}_D = \mathbf{T}_J \mathbf{R} \mathbf{T}_J^H, \quad (47)$$

$$\mathbf{w}_D = \frac{\mathbf{R}_D^{-1} \mathbf{s}_D(\theta)}{\mathbf{s}_D^H(\theta) \mathbf{R}_D^{-1} \mathbf{s}_D(\theta)}, \quad (48)$$

$$\mathbf{s}_D = \mathbf{T}_J \mathbf{s}(\theta_0), \quad (49)$$

where \mathbf{T}_J is defined in (45). The number of training samples for estimating \mathbf{B}_l is $N_B = 128$, averaged over 200 Montecarlo trials.

In Fig. 3 we simulated three jammers impinging on the ULA at angles -30° , 50° and 65° . All jammers have a JNR of -9 dB. One can see that for all decimation factors, F , the lowest SINR losses occur for filter lengths set as $L_v = F$, as we expected. The arrows in the figure point where the interpolator lengths are equal to the decimation factors. The presentation of the SINR loss in dB is equivalent to $10 \log_{10}(L_{\text{SINR}})$.

In Fig. 4 we simulated four jammers impinging on the ULA at angles -60° , -30° , 10° and 50° . All jammers have a high JNR of 15 dB. One can see that for all decimation factors, except for $F = 4$ (which is not the best reduction factor for this scenario anyway), the lowest SINR losses occur for filter

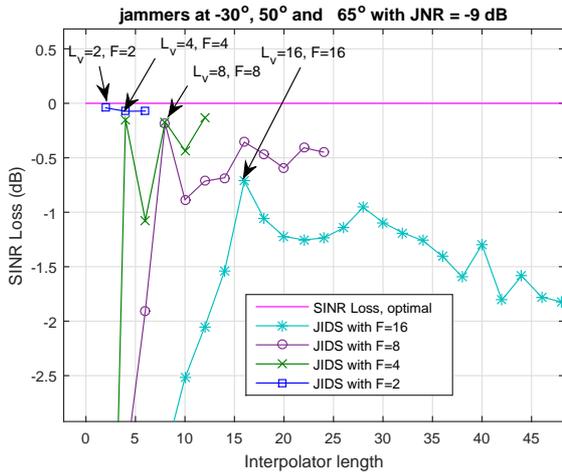


Fig. 3. SINR loss as a function of the filter lengths, L_v , for different decimation factor, F , for an array with $M = 64$ elements, $\text{SNR} = 10$ dB and 3 jammers with $\text{JNR} = -9$ dB.

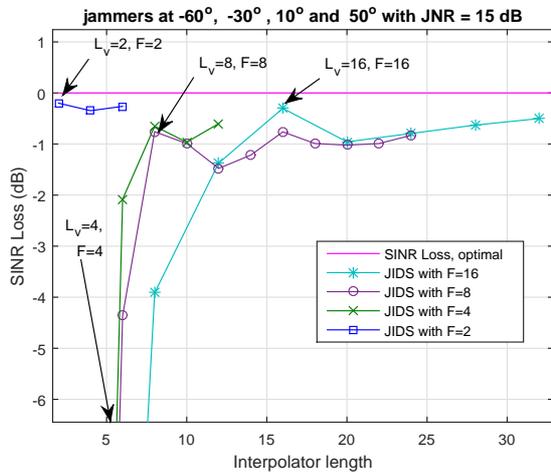


Fig. 4. SINR loss as a function of the filter lengths, L_v , for different decimation factor, F , for an array with $M = 64$ elements, $\text{SNR} = 10$ dB and 4 jammers with $\text{JNR} = 15$ dB.

lengths $L_v = F$. The arrows in the figure point where the interpolator lengths are equal to the decimation factors.

This procedure avoids the additional burden of jointly optimizing the filter length and the decimation factor.

C. JIDS Simplification for Beamforming Environment - JIDSB

Deeper investigation of the ULA structure into the JIDS revealed that the JIDS can be further simplified. In this context, we propose a low complexity criterium for selecting the best decimation pattern for ULAs. Considering the structure of the steering vector in a ULA, we can, instead of selecting the decimation pattern related to the largest eigenvalue, $\lambda_{\max, l}$, of \mathbf{F}_l , among all decimation patterns, $l \in \{0, \dots, F-1\}$, select the decimation pattern, l^* , that corresponds to the largest trace of \mathbf{F}_l , denoted $\text{tr}(\mathbf{F}_l)$,

$$l^* = \arg \max_l \{ \text{tr}(\mathbf{F}_l) \}, \quad (50)$$

$$l \in \{0, \dots, F-1\}. \quad (51)$$

This procedure leads to similar performance and avoids the need of eigenvalue decompositions during the decimation pattern decision process. The trace of \mathbf{F}_l (which is equal to the sum of all eigenvalues of \mathbf{F}_l) is approximately equal to the largest eigenvalue of \mathbf{F}_l , because, when using the length of the interpolation filter L_v equal to the reduction factor F , the rank of \mathbf{F}_l is at most two, meaning that \mathbf{F}_l has at most two non-zero eigenvalues.

Proof. The rank of $\mathbf{F}_l = \mathbf{B}_l^{-1} \mathbf{A}_l$ is upper bounded by the minimum between the rank of \mathbf{B}_l^{-1} and \mathbf{A}_l . Since \mathbf{B}_l is invertible, \mathbf{B}_l is a full rank matrix, leaving us with the analysis of \mathbf{A}_l . Matrix \mathbf{A}_l can be written as

$$\mathbf{A}_l = \mathbf{A}_{D_l}^H \mathbf{A}_{D_l}, \quad (52)$$

where

$$\mathbf{A}_{D_l} = \mathbf{D}_l \mathcal{S}. \quad (53)$$

The application of the JIDS in beamforming allows us to use the structure of the steering vector $\mathbf{s}(\theta_0) = [s_0, s_1, \dots, s_{M-1}]^T$ to go deeper into the structure of \mathbf{A}_{D_l} . The matrix $\mathcal{S} \in \mathbb{C}^{M \times L_v}$ is a Toeplitz matrix,

$$\mathcal{S} = \begin{pmatrix} s_0 & 0 & \dots & 0 \\ s_1 & s_0 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ s_{L_v-1} & s_{L_v-2} & \dots & s_0 \\ \vdots & \vdots & \ddots & \vdots \\ s_{M-1} & s_{M-2} & \dots & s_{M-L_v} \end{pmatrix}, \quad (54)$$

with its element, $\mathcal{S}_{p,q}$, at the p -th row, $p \in \{0, \dots, M-1\}$, and q -th column, $q \in \{0, \dots, L_v-1\}$, formed by

$$\mathcal{S}_{p,q} = \begin{cases} s_{p-q}, & p \geq q \\ 0, & \text{otherwise.} \end{cases} \quad (55)$$

The m -th element, s_m , of the steering vector $\mathbf{s}(\theta_0) \in \mathbb{C}^{M \times 1}$ corresponds to the signal impinging on the m -th antenna element and is given by

$$s_m = e^{-j \frac{2\pi d}{\lambda_c} m \sin(\theta_0)}, \quad m \in \{0, \dots, M-1\}. \quad (56)$$

Substituting (56) into (55),

$$\mathcal{S}_{p,q} = \begin{cases} e^{-j \frac{2\pi d}{\lambda_c} (p-q) \sin(\theta_0)}, & p \geq q \\ 0, & \text{otherwise,} \end{cases} \quad (57)$$

or equivalently

$$\mathcal{S}_{p,q} = \begin{cases} e^{\alpha p} e^{-\alpha q}, & p \geq q \\ 0, & \text{otherwise,} \end{cases} \quad (58)$$

where

$$\alpha = -j \frac{2\pi d}{\lambda_c} \sin(\theta_0). \quad (59)$$

We can then rewrite the ‘‘tall’’ matrix \mathcal{S} as

$$\mathcal{S} = \begin{pmatrix} e^{\alpha 0} e^{-\alpha 0} & 0 & \dots & 0 \\ e^{\alpha 1} e^{-\alpha 0} & e^{\alpha 1} e^{-\alpha 1} & \dots & 0 \\ \vdots & \vdots & \vdots & \vdots \\ e^{\alpha(L_v-1)} e^{-\alpha 0} & e^{\alpha(L_v-1)} e^{-\alpha 1} & \dots & e^{\alpha(L_v-1)} e^{-\alpha(L_v-1)} \\ \vdots & \vdots & \vdots & \vdots \\ e^{\alpha(M-1)} e^{-\alpha 0} & e^{\alpha(M-1)} e^{-\alpha 1} & \dots & e^{\alpha(M-1)} e^{-\alpha(L_v-1)} \end{pmatrix}. \quad (60)$$

Matrix $\mathcal{S} \in \mathbb{C}^{M \times L_v}$ has at most L_v linearly independent rows. Indeed, from (60), we note that the $M - L_v$ last rows of \mathcal{S} are linearly dependent: they can be expressed as the multiplication of the row vector, $[e^{-\alpha 0}, e^{-\alpha 1}, \dots, e^{-\alpha(L_v-1)}]$, by a complex scalar. This means that the L_v linearly independent rows of \mathcal{S} are precisely the first L_v rows of \mathcal{S} .

After decimation using the l -th uniform pattern, the i -th row of the resulting matrix \mathbf{A}_{D_l} , denoted $\mathbf{A}_{D_l}(i, :)$, corresponds to the $(iF + l)$ -th row of matrix \mathcal{S} , denoted $\mathcal{S}(iF + l, :)$,

$$\mathbf{A}_{D_l}(i, :) = \mathcal{S}(iF + l, :), \quad (61)$$

$$i \in \{0, \dots, D - 1\} \quad (62)$$

$$l \in \{0, \dots, F - 1\}. \quad (63)$$

Using $L_v = F$, the l -th uniform decimation pattern, $l \in [0, \dots, F - 2]$ has two linearly independent rows, while decimation pattern $l = F - 1$ has only one linearly independent row. Therefore, the rank of \mathbf{F}_l is limited by the rank of \mathbf{A}_l which in turn is limited by the rank of \mathbf{A}_{D_l} , consequently the rank is at most two. This finishes the proof that \mathbf{F}_l has at most two nonzero eigenvalues. \square

As a result, we can choose the \mathbf{F}_l , $l \in \{0, \dots, F - 1\}$, which has the largest trace (which is equal to the sum of all eigenvalues), instead of the one that has the largest eigenvalue without any noticeable performance degradation.

To compute the trace of \mathbf{F}_l , we can use the fact that the trace of a matrix $\mathbf{C} = \mathbf{A}\mathbf{B}$, with $\mathbf{A}, \mathbf{B}, \mathbf{C} \in \mathbb{C}^{N \times N}$, is given by $\text{tr}(\mathbf{C}) = \sum_{i=0}^{N-1} \mathbf{A}(i, :)\mathbf{B}(:, i)$, where $\mathbf{A}(i, :)$ denotes the i -th row of \mathbf{A} and $\mathbf{B}(:, i)$ denotes the i -th column of \mathbf{B} . Thus the trace of \mathbf{F}_l is computed as

$$\text{tr}(\mathbf{F}_l) = \sum_{i=0}^{L_v-1} \mathbf{B}_l^{-1}(i, :)\mathbf{A}_l(:, i). \quad (64)$$

A representative result is depicted in Fig. 5, showing that the proposed decimation pattern selection procedure is in good agreement with the original one. In Fig. 5 we simulate a ULA with $M = 64$ elements; SOI at 0° , SNR = 10 dB; 8 jammers impinging at angles $-80^\circ, -65^\circ, -40^\circ, -25^\circ, 30^\circ, 45^\circ, 60^\circ$ and 75° with a JNR of 15 dB each. We selected the interpolator length, $L_v = F$, as described in subsection IV-B.

The JIDS algorithm specialized for beamforming with the proposed simplification is named the JIDSB dimensionality reduction algorithm.

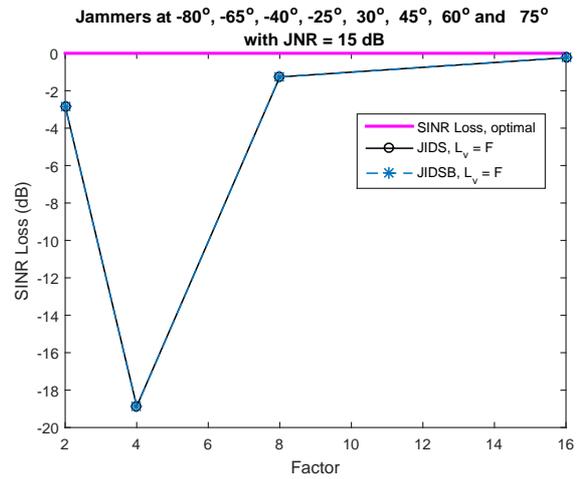


Fig. 5. SINR loss as a function of the decimation factor, F , for the proposed and the optimal decimation pattern selection strategies for an array with $M = 64$ elements, SNR = 10 dB and 8 jammers with JNR = 15 dB.

D. Computational Complexity

In this subsection, we address the computational complexity of the JIDS and the JIDSB algorithms.

The main steps of the proposed algorithms take place in a lower dimensional subspace, because the practical effect of the decimation matrix \mathbf{D}_l is to select just D lines of \mathcal{S} and $\mathcal{R}(i)$. Thus, the computation of matrices $\mathbf{A}_l = \mathcal{S}^H \mathbf{D}_l^H \mathbf{D}_l \mathcal{S}$ and $\hat{\mathbf{B}}_l(i) = \mathcal{R}^H(i) \mathbf{D}_l^H \mathbf{D}_l \mathcal{R}(i)$ are reduced to the multiplication of two matrices of size $D \times L_v$. Tables I and II show the computational complexity of the main parts of the JIDS and JIDSB algorithms respectively.

Fig. 6 shows how the computational complexity of the stage of decimation pattern selection is decreased by the simplification described in subsection IV-C as a function of the reduction factor F . In order to assess the number of operations required for finding the eigenvector associated with the largest eigenvalue of a $N \times N$ matrix we used the power method [20], which takes N_{it} iterations and involves $N_{it}N^2$ complex multiplications and $N_{it}N(N - 1)$ complex additions. For both algorithms (JIDS and JIDSB), we set the filter length equal to the reduction factor $L_v = F$ and $N_{it} = 5$. We can see that the proposed simplification significantly reduces the number of complex operations for the stage of decimation pattern selection.

V. NUMERICAL RESULTS AND COMPARISONS

In this section, we compare the JIDSB algorithm applied for reducing the dimensionality of the MVDR-SMI beamformer (JIDSB-SMI) with renowned rank reduction algorithms in terms of SINR loss performance, computational complexity, adapted beampattern and BER performance.

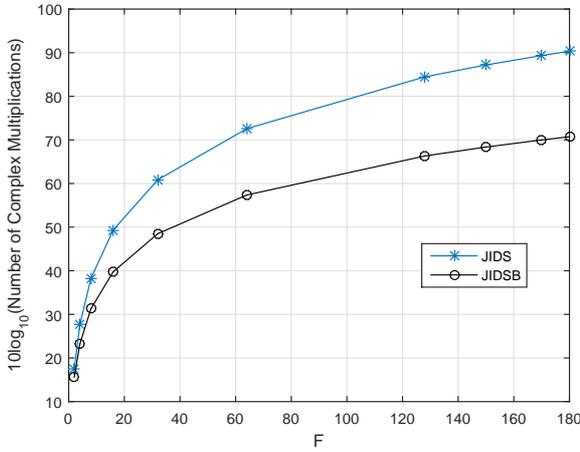
We compare the JIDSB-SMI with the PC-SMI, CSM-SMI and CG-MWF algorithms. The PC-SMI and CSM-SMI algorithms follow the same structure of the JIDS-SMI as explained in section III: the received array snapshots go through the dimensionality reduction stage and then are used to estimate the lower dimension autocorrelation matrix which is then

TABLE I
 COMPLEX OPERATIONS OF THE JIDS.

Algorithm main steps	Multiplications	Additions
Computation of \mathbf{A}_l	FDL_v^2	$F(D-1)L_v^2$
Estimation of \mathbf{B}_l	$N_B FDL_v^2 + FL_v^2$	$N_B F(D-1)L_v^2 + (N_B - 1)FL_v^2$
Inversion of \mathbf{B}_l	$O(FL_v^3)$	$O(FL_v^3)$
Computation of \mathbf{F}_l	FL_v^3	$F(L_v - 1)L_v^2$
Eigenvalue decomposition	$O(FL_v^3)$	$O(FL_v^3)$

 TABLE II
 COMPLEX OPERATIONS OF THE JIDSB.

Algorithm main steps	Multiplications	Additions
Computation of \mathbf{A}_l	FDL_v^2	$F(D-1)L_v^2$
Estimation of \mathbf{B}_l	$N_B FDL_v^2 + FL_v^2$	$N_B F(D-1)L_v^2 + (N_B - 1)FL_v^2$
Inversion of \mathbf{B}_l	$O(FL_v^3)$	$O(FL_v^3)$
Computation of $tr(\mathbf{F}_l)$	FL_v^2	$F(L_v^2 - 1)$
Computation of \mathbf{F}_l	L_v^3	$(L_v - 1)L_v^2$
Eigenvalue decomposition	$O(L_v^3)$	$O(L_v^3)$


 Fig. 6. Comparison of the number of complex multiplications of the stage of decimation pattern selection of the JIDS and JIDSB for $L_v = F$ and $N_{it} = 5$.

inverted and used in the computation of the MVDR or the MPDR filter. The rank of the PC-SMI algorithm is the number of vectors used to form the subspace that is spanned by the D eigenvectors associated with the D largest eigenvalues of the estimated autocorrelation matrix $\hat{\mathbf{R}}$ [15]. The rank of the CSM-SMI algorithm is the number of vectors used to form the subspace that is spanned by the D eigenvectors of the estimated autocorrelation matrix $\hat{\mathbf{R}}$ that maximize the cross spectral metric [17]. The rank of the CG-MWF algorithm is the number of basis vectors used to describe the Krylov subspace of the estimated autocorrelation matrix $\hat{\mathbf{R}}$ [22]. The CG-MWF algorithm converges to the MVDR or the MPDR result without the need of inverting the estimated autocorrelation matrix $\hat{\mathbf{R}}$. The rank of the JIDSB-SMI is $\lfloor M/F \rfloor = D$, which is the final length of the vectors after the dimensionality reduction stage.

Tables III, IV, V and VI show step by step how the JIDSB-SMI, PC-SMI, CSM-SMI and CG-MWF algorithms implement the MVDR beamformer respectively.

 TABLE III
 JIDSB-SMI ALGORITHM

initialization $\hat{\mathbf{R}}_D(0) = \mathbf{0}_{D \times D}$, F $\mathcal{S} = \text{Toeplitz}(\mathbf{s}(\theta_0))$
JIDSB for dimensionality reduction for $l = 1, \dots, F$ $\mathbf{p}_l = l$ -th fixed decimation pattern, $\mathbf{A}_l = \mathcal{S}^H \text{diag}(\mathbf{p}_l) \mathcal{S}$, $\hat{\mathbf{B}}_l = \frac{1}{N_B} \sum_{i=1}^{N_B} \mathcal{R}^H(i) \text{diag}(\mathbf{p}_l) \mathcal{R}(i)$, where $\mathcal{R}(i) = \text{Toeplitz}(\mathbf{x}(i))$ end $l^* = \arg \max_l \sum_{i=1}^{L_v} \mathbf{B}_l^{-1}(i, :) \mathbf{A}_l(:, 1)$, $\hat{\mathbf{F}}_{l^*} = \hat{\mathbf{B}}_{l^*}^{-1} \mathbf{A}_{l^*}$ \mathbf{v} = eigenvector associated with the largest eigenvalue of $\hat{\mathbf{F}}_{l^*}$ $\mathcal{V} = \text{Toeplitz}(\mathbf{v})$ \mathbf{D} is constructed according to (13)
MVDR-SMI after dimensionality reduction stage $\mathbf{T}_D = \mathbf{D} \mathcal{V}$ $\mathbf{s}_D = \mathbf{T}_D \mathbf{s}(\theta_0)$ $\hat{\mathbf{R}}_D = \frac{1}{N_s} \sum_{i=1}^{N_s} \mathbf{T}_D \mathbf{r}(i) \mathbf{r}^H(i) \mathbf{T}_D^H = \mathbf{T}_D \hat{\mathbf{R}} \mathbf{T}_D^H$ $\mathbf{w}_D = \hat{\mathbf{R}}_D^{-1} \mathbf{s}_D / \mathbf{s}_D^H \hat{\mathbf{R}}_D^{-1} \mathbf{s}_D$

 TABLE IV
 PC-SMI MVDR ALGORITHM

initialization $\hat{\mathbf{R}}(0) = \mathbf{0}_{M \times M}$, $D_{PC} = D$ $\hat{\mathbf{R}} = \frac{1}{N_s} \sum_{i=1}^{N_s} \mathbf{x}(i) \mathbf{x}^H(i)$
PC rank reduction $\mathbf{T}_{PC} = [\phi_1 \dots \phi_{D_{PC}}]$ ϕ_i = eigenvector associated with the i -th largest eigenvalue of $\hat{\mathbf{R}}$
MVDR-SMI after rank reduction stage $\mathbf{s}_{PC} = \mathbf{T}_{PC}^H \mathbf{s}(\theta_0)$ $\hat{\mathbf{R}}_{PC} = \mathbf{T}_{PC}^H \hat{\mathbf{R}} \mathbf{T}_{PC}$ $\mathbf{w}_{PC} = \hat{\mathbf{R}}_{PC}^{-1} \mathbf{s}_{PC} / \mathbf{s}_{PC}^H \hat{\mathbf{R}}_{PC}^{-1} \mathbf{s}_{PC}$

A. SINR Loss Performance Comparison

We evaluate the SINR loss performance of the proposed JIDSB-SMI algorithm and compare it with the performance of the JIDS-SMI, PC-SMI, CSM-SMI and the CG-MWF algorithm.

In these simulations we adopt a ULA consisting of $M = 64$ sensor elements whose inter-element spacing is half a signal wavelength. The SOI is at 0° . We simulate two scenarios

TABLE V
 CSM-SMI MVDR ALGORITHM

initialization $\hat{\mathbf{R}}(0) = \mathbf{0}_{M \times M}$, $D_{\text{CSM}} = D$ $\hat{\mathbf{R}} = \frac{1}{N_s} \sum_{i=1}^{N_s} \mathbf{x}(i)\mathbf{x}^H(i)$
CSM rank reduction $\mathbf{T}_{\text{CSM}} = [\phi_1 \dots \phi_{D_{\text{CSM}}}]$ $\phi_i =$ eigenvector associated with the i -th largest S_j $S_j = \frac{\phi_j^H \mathbf{s}(\theta_0)}{\lambda_j}$, $j \in [1, \dots, M]$ $\phi_j =$ eigenvector associated with the j -th eigenvalue λ_j of $\hat{\mathbf{R}}$
MVDR-SMI after rank reduction stage $\mathbf{s}_{\text{CSM}} = \mathbf{T}_{\text{CSM}}^H \mathbf{s}(\theta_0)$ $\hat{\mathbf{R}}_{\text{CSM}} = \mathbf{T}_{\text{CSM}}^H \hat{\mathbf{R}} \mathbf{T}_{\text{CSM}}$ $\mathbf{w}_{\text{CSM}} = \hat{\mathbf{R}}_{\text{CSM}}^{-1} \mathbf{s}_{\text{CSM}} / \mathbf{s}_{\text{CSM}}^H \hat{\mathbf{R}}_{\text{CSM}}^{-1} \mathbf{s}_{\text{CSM}}$

 TABLE VI
 CG-MWF ALGORITHM

initialization $\mathbf{u}_1 = -\mathbf{g}_0 = \mathbf{s}(\theta_0)$, $\gamma_0 = \ \mathbf{g}_0\ ^2$, $\mathbf{w}_0 = \mathbf{0}_{M \times 1}$, $K = D$ $\hat{\mathbf{R}} = \frac{1}{N_s} \sum_{i=1}^{N_s} \mathbf{x}(i)\mathbf{x}^H(i)$
CG-MWF for each iteration $k = 1, \dots, K$ $\mathbf{v}_k = \hat{\mathbf{R}} \mathbf{u}_k$ $\eta_k = \frac{\gamma_{k-1}}{\mathbf{u}_k^H \mathbf{v}_k}$ $\mathbf{w}_k = \mathbf{w}_{k-1} + \eta_k \mathbf{u}_k$ $\mathbf{g}_k = \mathbf{g}_{k-1} + \eta_k \mathbf{v}_k$ $\gamma_k = \ \mathbf{g}_k\ ^2$ $\mathbf{u}_{k+1}(i) = -\mathbf{g}_k(i) + \frac{\gamma_{k-1}}{\gamma_k} \mathbf{u}_k$
Output: $\mathbf{w}_{\text{CG-MWF}} = \frac{\mathbf{w}_K}{\mathbf{s}^H(\theta_0) \mathbf{w}_K}$

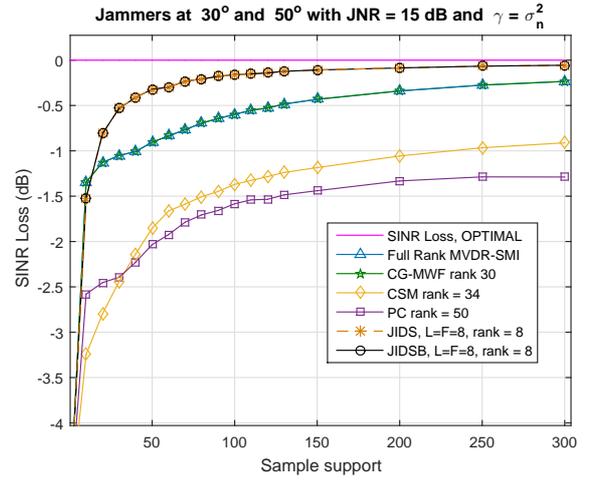
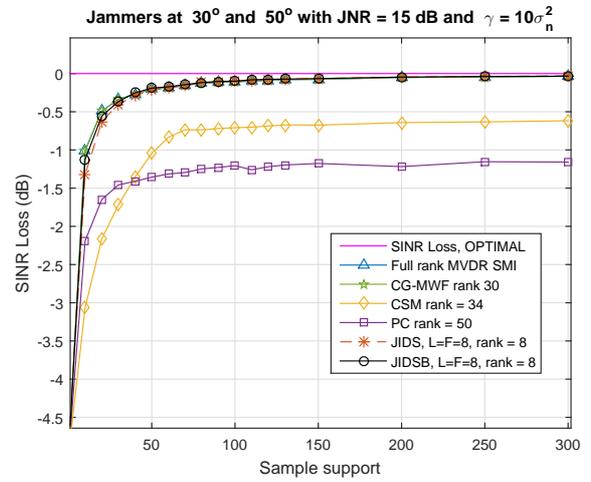
and compare the algorithms SINR loss for different sample supports. In the first scenario, there are two narrowband jammers impinging at angles 30° and 50° and in the second scenario, there are 6 narrowband jammers impinging at angles -65° , -40° , -25° , 30° , 45° and 60° . Each jammer has a JNR of 15 dB.

For both scenarios, we examine the case when the amount of diagonal loading is set to $\gamma = \sigma_n^2$ in (9), just in order to avoid computational instabilities in the algorithms and the case when the diagonal loading is set to $\gamma = 10\sigma_n^2$, which is empirically shown in [34] to be a suitable value. We compare the latter case using two different detection filters: the MVDR, when the desired signal is not present during estimation of the autocorrelation matrix and the MPDR, when the desired signal is present during the estimation of the autocorrelation matrix and the SOI SNR equals 10 dB.

For all simulations the JIDS and JIDSB had equivalent performances, as expected, so in the following we will mention only the JIDSB for brevity. The mean SINR loss is defined as $L_{\text{SINR}} = \text{SINR}/\text{SNR}_0$, with $\text{SNR}_0 = \mathbf{s}^H(\theta_0) \mathbf{R}^{-1} \mathbf{s}(\theta_0)$, where \mathbf{R} is the true autocorrelation matrix of noise and interference. The SINR is computed as

$$\text{SINR} = \frac{\sigma_0^2 \mathbb{E} \{ |\mathbf{w}^H \mathbf{T} \mathbf{s}(\theta_0)|^2 \}}{\mathbb{E} \{ \mathbf{w}^H \mathbf{T} \mathbf{R} \mathbf{T}^H \mathbf{w} \}}, \quad (65)$$

where \mathbf{w} is the beamforming filter, \mathbf{T} is the dimensionality reduction transformation (applied when necessary) and the expectation, $\mathbb{E}\{\cdot\}$, is computed by averaging 200 Montecarlo runs. Figs 7 and 10 show the SINR loss vs. sample support with a diagonal loading of σ_n^2 . We can see that the JIDSB


 Fig. 7. Output SINR loss versus the sample support with $M = 64$, 2 jammers with JNR = 15 dB and $\gamma = \sigma_n^2$.

 Fig. 8. Output SINR loss versus the sample support with $M = 64$, 2 jammers with JNR = 15 dB and $\gamma = 10\sigma_n^2$.

had an expressive superior performance compared to full rank MVDR-SMI, CG-MWF, CSM-SMI and PC-SMI algorithms.

Figs. 8 and 11 show the SINR loss vs. sample support with a diagonal loading of $10\sigma_n^2$. As expected, all the algorithms improved their convergence rate when compared to the case of a diagonal loading of σ_n^2 depicted in Figs 7 and 10. But, even with the considerable improvement of the full rank MVDR-SMI and CG-MWF, the proposed JIDSB still had similar performance.

Next, we examine the effect of the presence of the desired signal in the observed data during estimation of the autocorrelation matrix. Figs 9 and 12 show how the performance is affected by the self-nulling when the MPDR detection filter is applied. As expected, for a high SNR such as 10 dB, there is a large degradation in the performance of all algorithms. Still, the JIDSB performed better than all the others algorithms. It should be stressed that in Fig. 9 the JIDSB showed an impressive superior performance.

Finally, we evaluate the SINR loss performance vs. number of antennas for all the algorithms considered, for a sample

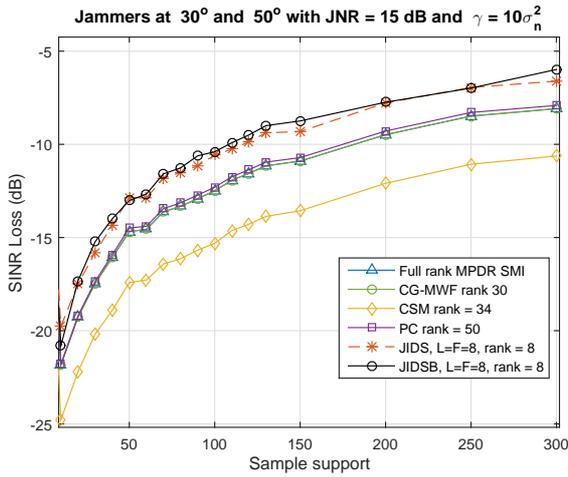


Fig. 9. Output SINR loss versus the sample support with $M = 64$, 2 jammers with $JNR = 15$ dB and $\gamma = 10\sigma_n^2$ when the desired signal is present during estimation of the autocorrelation matrix and $SNR = 10$ dB.

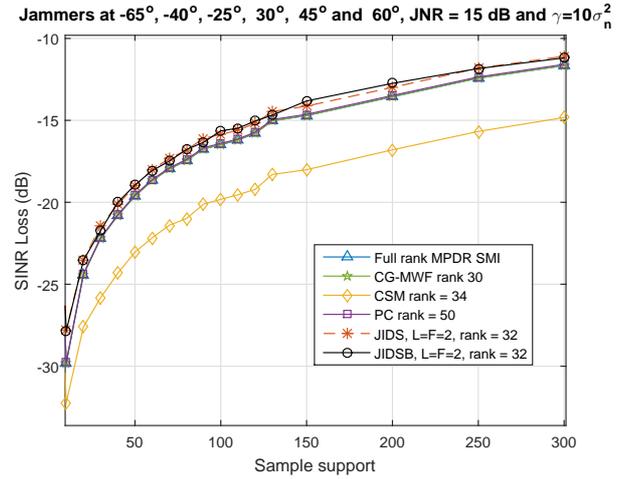


Fig. 12. Output SINR loss versus the sample support with $M = 64$, 6 jammers with $JNR = 15$ dB and $\gamma = 10\sigma_n^2$ when the desired signal is present during estimation of the autocorrelation matrix and $SNR = 10$ dB.

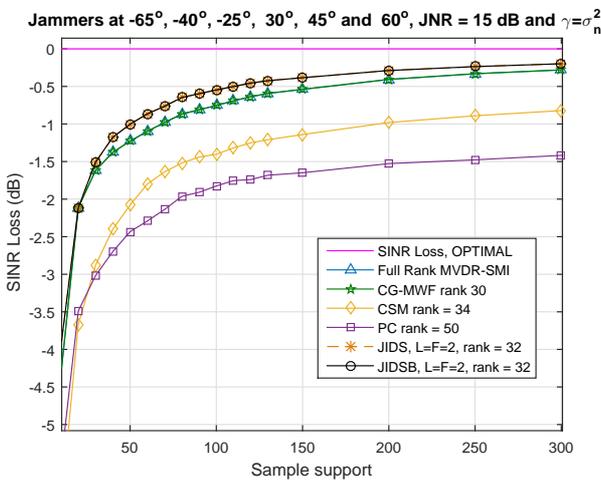


Fig. 10. Output SINR loss versus the sample support with $M = 64$, 6 jammers with $JNR = 15$ dB and $\gamma = \sigma_n^2$.

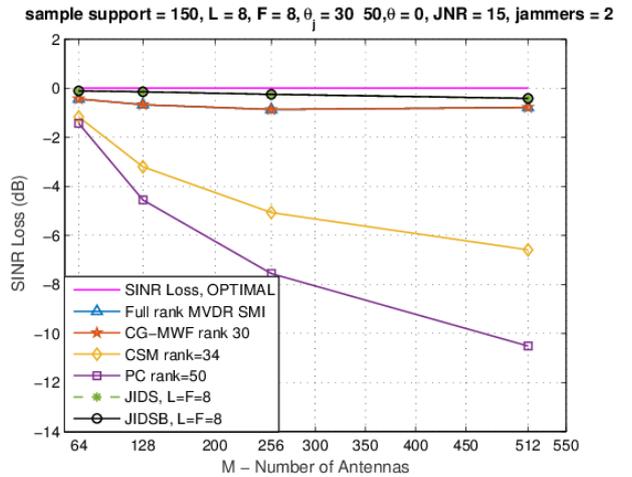


Fig. 13. Output SINR loss versus number of antennas for different algorithms for sample support $N_s = 150$, 2 jammers with $JNR = 15$ dB.

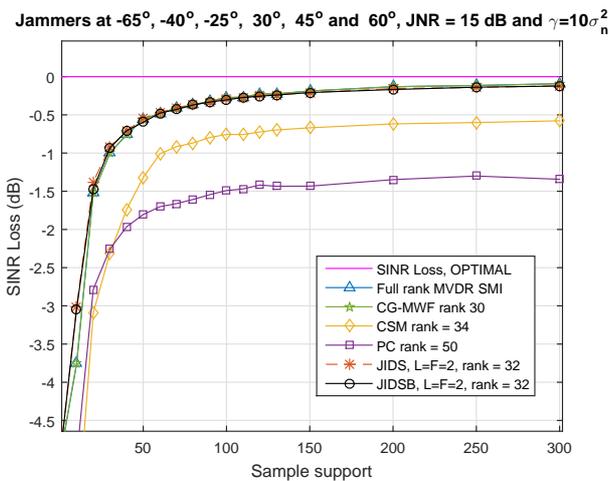


Fig. 11. Output SINR loss versus the sample support with $M = 64$, 6 jammers with $JNR = 15$ dB and $\gamma = 10\sigma_n^2$.

support of 150 snapshots and two jammers. As can be seen from Fig. 13, as the number of antennas M grows (and consequently the dimensionality of the snapshot increases), the SINR loss performances of the CSM-SMI and the PC-SMI algorithms degrade rather abruptly. Still, the CG-MWF and JIDSB algorithms exhibit only a slight degradation in SINR loss performance.

B. Computational Complexity Comparison

Table VII shows the number of complex operations needed to complete all the steps described in tables III, IV, V and VI for the algorithms considered. We used the power method (*cf* Section IV-D) in order to find the eigenvector associated with the largest eigenvalue of a $N \times N$ matrix; the number of iterations $N_{it} = 5$. In order to compute the inverse of a $N \times N$ matrix we used the Gauss-Jordan method that takes $2N^3/6 + 3N^2/6 - 5N/6$ complex multiplications and additions.

Fig. 14 depicts a comparison of the number of complex multiplications computed by the different algorithms for the

TABLE VII
COMPLEX OPERATIONS OF THE JISDB-SMI, PC-SMI, CSM-SMI AND CG-MWF ALGORITHMS.

Algorithm	Multiplications	Additions
JISDB-SMI	$(N_s + 1)DM + D^2(N_s + 2) + 2D$ $+ 2D^3/6 + 3D^2/6 - 5D/6$ $+ FDL_v^2(1 + N_B) + L_v^2(2F + N_{it}) + L_v^3$ $+ 2FL_v^3/6 + 3FL_v^2/6 - 5FL_v/6$	$N_s D(M - 1) + N_s D^2 + D(M - 1) - 1$ $+ 2D^3/6 + 3D^2/6 - 5D/6$ $+ FDL_v^2(1 + N_B) - L_v^2(F + 1) - F + N_{it}L_v(L_v - 1)$ $+ L_v^3 + 2FL_v^3/6 + 3FL_v^2/6 - 5FL_v/6$
PC-SMI	$(N_s + 1)M^2 + D_{PC}N_{it}M^2 + (1 + M)(MD_{PC} + D_{PC}^2)$ $+ 2D_{PC} + 2D_{PC}^3/6 + 3D_{PC}^2/6 - 5D_{PC}/6$	$(N_s - 1)M^2 + D_{PC}(M - 1)(N_{it}M + M + D_{PC} + 1)$ $+ D_{PC}^2 - 1 + 2D_{PC}^3/6 + 3D_{PC}^2/6 - 5D_{PC}/6$
CSM-SMI	$(N_s + 1)M^2 + N_{it}M^3$ $+ (M + 1)(M + MD_{CSM} + D_{CSM}^2)$ $+ 2D_{CSM} + 2D_{CSM}^3/6 + 3D_{CSM}^2/6 - 5D_{CSM}/6$	$(N_s - 1)M^2 + (M - 1)(1 + D_{CSM})(M + D_{CSM}) - 1$ $+ N_{it}(M - 1)M^2 + D_{CSM}^2 + 2D_{CSM}^3/6$ $+ 3D_{CSM}^2/6 - 5D_{CSM}/6$
CG-MWF	$(N_s + 1)M^2 + D_{MWF}(M^2 + 5M + 2) + 4M$	$M^2(N_s - 1) + D_{MWF}(M^2 + 4M - 2) + 2(M - 1)$

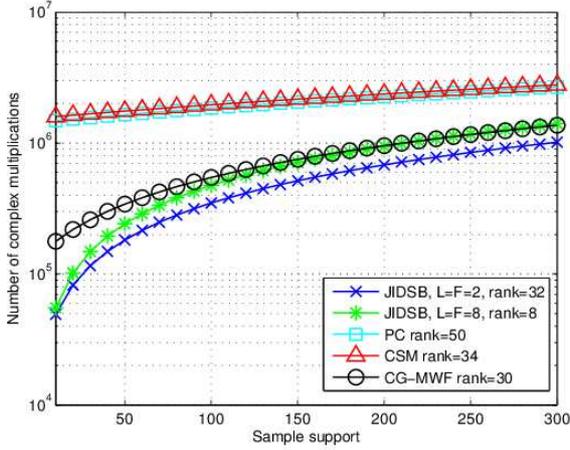


Fig. 14. Comparison of the number of complex multiplications in logarithmic scale for different algorithms for $M = 64$.

two scenarios considered in section V-A, according to Table VII. Analysing Fig. 14 in light of figs. 7, 8, 9, 10, 11 and 12, it can be seen that the superior SINR loss performance of the JISDB compared to the others algorithms does not come at the expense of increased computational complexity. Indeed, the JISDB performs better than the CSM-SMI and the PC-SMI algorithms and still has lower computational complexity.

Even for the considerable performance improvement of the CG-MWF in the case of diagonal loading, the proposed JISDB still had similar performance, with a significantly lower computational complexity than the CG-MWF for the case of Fig. 11 ($L=F=2$) and a lower computational complexity than the CG-MWF for smaller sample support and similar computational complexity as the CG-MWF for larger sample support for the case of Fig. 8 ($L=F=8$), as can be verified in Fig. 14.

Since the main contender of the JISDB in terms of SINR loss performance is the CG-MWF, in Fig. 15 we further compare the computational complexity of the JISDB algorithm with the CG-MWF for different decimation factors F , vs. different sizes of sample support N_s . The number of snapshots N_B used to estimate $\mathbf{B}_l \in \mathbb{C}^{L_v \times L_v}$ is the total number of snapshots available, $N_B = N_s$. We note from Fig. 15 that the JISDB has a remarkably smaller computational complexity than the CG-MWF for factors $F = 2$ and $F = 4$. For $F = 8$,

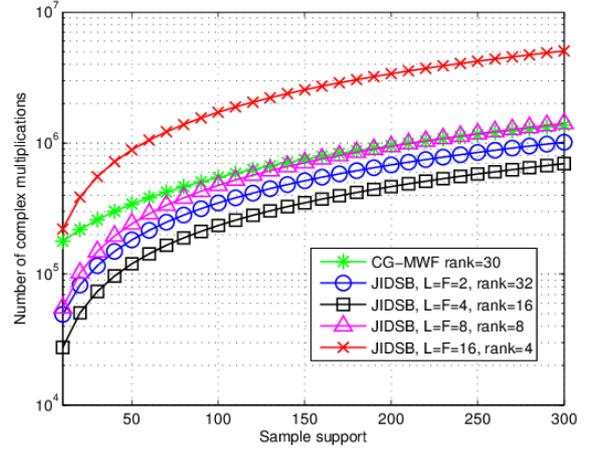


Fig. 15. Comparison of the number of complex multiplications of the JISDB with the CG-MWF as a function of the sample support, N_s , for $F = \{2, 4, 8, 16\}$, $M = 64$, $N_{it} = 5$, $D_{MWF} = 30$ and $N_B = N_s$.

the complexity is notably lower for smaller sample support and it is almost the same as the computational complexity of the CG-MWF for larger sample support. For $F = 16$, the JISDB has a significantly increased computational complexity, that is because for this pair of array size M and factor F , the JISDB pre-processing steps take place in a not so reduced subspace, thus increasing significantly the computational complexity.

C. Beam pattern Comparison

Fig. 16 depicts the adapted beampattern of all algorithms with the configuration that led to the results in Fig. 11 with a sample support of 200 snapshots. The arrows indicate the positions of the jammers. All curves were averaged over 200 Montecarlo runs. From Fig. 16, we can see that all algorithms were able to set deep nulls at the jammers position. Moreover, the JISDB beampattern follows more closely the beampatterns obtained by the CG-MWF and the full rank MVDR-SMI, which is in good agreement with the SINR loss performance behaviour depicted in Fig. 11.

D. BER Performance Comparison

Fig. 17 illustrates the application of the sensor array for a communication system and depicts the BER, for all algorithms for a small sample support. The system model is as described

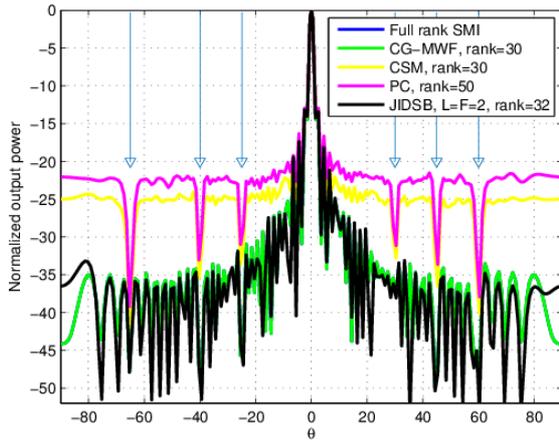


Fig. 16. Adapted beampattern for different algorithms for sample support $N_s = 200$, $M = 64$ and 6 jammers with $JNR = 15$ dB.

in (1) and the scenario is again the same related to Fig. 11. The simulated desired signal, b_0 , represents a symbol of a Quadrature Phase Shift Keying (QPSK) constellation with power $\sigma_0^2 = E[|b_0|^2]$ and impinges on the array from direction $\theta_0 = 0^\circ$. The outputs of the beamformer filters, $z(i)$ as in (4) and $z_D(i)$ as in (11), are fed to a minimum-distance QPSK detector, which is the optimal detector for gaussian channels.

Since noise and jammers are modeled as independent complex gaussian random vectors, it is possible to compute semi-analytically the error bit probability, $P(e)$, as

$$P(e) = E[P(e|\text{SINR}(\mathbf{w}))], \quad (66)$$

which can be estimated by the average

$$\text{BER} = \frac{1}{N_p} \sum_{i=1}^{N_p} P(e|\text{SINR}(\mathbf{w}_i)), \quad (67)$$

where

$$P(e|\text{SINR}(\mathbf{w}_i)) = Q\left(\sqrt{\text{SINR}(\mathbf{w}_i)}\right), \quad (68)$$

with

$$Q(\alpha) = \int_{\alpha}^{\infty} \frac{1}{\sqrt{2\pi}} e^{-\frac{t^2}{2}} dt, \quad (69)$$

and

$$\text{SINR}(\mathbf{w}_i) = \frac{\sigma_0^2 \|\mathbf{w}_i^H \mathbf{T}_i \mathbf{s}(\theta_0)\|^2}{\mathbf{w}_i^H \mathbf{T}_i \mathbf{R} \mathbf{T}_i^H \mathbf{w}_i}, \quad (70)$$

where \mathbf{w}_i and \mathbf{T}_i are the detection filter and the dimensionality reduction transformation obtained in the i -th simulation run and \mathbf{R} is the autocorrelation matrix of the noise and interference. Fig. 17 depict the semi-analytical BER of the described QPSK system using $N_p = 1000$ for a sample support of 20. In the horizontal axis we show the SNR based on the reference detection SNR of the optimal full rank case without interference, given as

$$\text{SNR} = 10 \log_{10} \left(\frac{\sigma_0^2}{\sigma_n^2} \|\mathbf{s}(\theta_0)\|^2 \right) = 10 \log_{10} \left(\frac{M\sigma_0^2}{\sigma_n^2} \right). \quad (71)$$

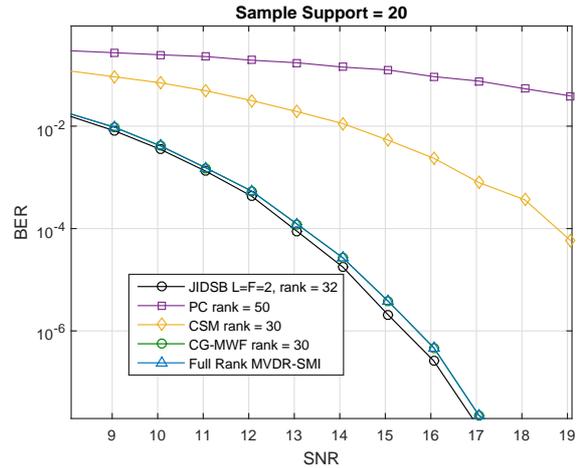


Fig. 17. Semi-analytical BER for different algorithms for sample support $N_s = 20$, $M = 64$, 6 jammers with $JNR = 15$ dB.

We can see that the JIDSB performance is slightly better than the full rank MVDR-SMI and CG-MWF, that is because the mean SINR for that scenario, with a sample support of only 20 snapshots is slightly better for the JIDSB than for the CG-MWF.

VI. CONCLUSION

We reported the JIDS dimensionality reduction technique. The JIDS algorithm achieves dimensionality reduction by means of a joint interpolation and decimation scheme. The JIDS strategy design is an elegant and effective way to obtain the joint interpolation filter and decimation pattern, taking advantage of the correlation generated by the interpolation filter in order to eliminate samples and still achieve a high SINR after the decimation stage. We stress that the design is such that the interpolator filter maximizes the SINR at the output of the decimation stage, irrespective of the final application filter. We also proposed the JIDSB algorithm, which is a specialized version of the JIDS technique for beamforming with new proposed simplification procedures that resulted from analysis of the combination of the beamforming system model with the JIDS structure. The JIDSB has reduced number of operations and complexity in comparison with the JIDS, without degrading its performance.

We presented performance results in terms of SINR loss vs. sample support applying the MVDR filter with and without diagonal loading and the MPDR filter with diagonal loading. We could see the superiority of the JIDSB-SMI in terms of SINR loss performance for the MPDR filter and for the MVDR filter without diagonal loading. In the case of the MVDR filter with diagonal loading the JIDSB had similar performance as the full rank MVDR-SMI and CG-MWF, with an impressive lower computational complexity specially for small sample supports.

In our understanding, the proposed JIDSB algorithm shows considerable advantages in beamforming scenarios. It is an inherently robust method, it has similar or superior SINR loss performance than the full rank MVDR-SMI and CG-MWF,

lower computational complexity and it reduces significantly the length of the processed data.

REFERENCES

- [1] R. E. Crochiere and L. R. Rabiner, *Multirate Digital Signal Processing*, Englewood Cliffs, NJ: Prentice-Hall, 1983.
- [2] R. C. de Lamare and R. Sampaio-Neto, "Adaptive reduced-rank MMSE filtering with interpolated FIR filters and adaptive interpolators," *IEEE Signal Process. Lett.*, vol. 12, no. 3, pp. 177–180, Mar. 2005, doi:10.1109/LSP.2004.842290.
- [3] R. C. de Lamare and R. Sampaio-Neto, "Reduced-rank interference suppression for DS-CDMA based on interpolated FIR filters," *IEEE Commun. Lett.*, vol. 9, no. 3, pp. 213–215, Mar. 2005, doi:10.1109/LCOMM.2005.03010.
- [4] R. C. de Lamare and R. Sampaio-Neto, "Adaptive interference suppression for DS-CDMA systems based on interpolated FIR filters with adaptive interpolators in multipath channels," *IEEE Trans. Veh. Technol.*, vol. 56, no. 5, pp. 2457–2474, Sep. 2007, doi:10.1109/TVT.2007.899931.
- [5] R. C. de Lamare and R. Sampaio-Neto, "Adaptive reduced-rank processing based on joint and iterative interpolation, decimation, and filtering," *IEEE Trans. Signal Process.*, vol. 57, no. 7, pp. 2503–2514, Jul. 2009, doi:10.1109/TSP.2009.2018641.
- [6] A. de Oliveira, "Detecção de sinais em sistema de comunicações de banda ultra-larga," Master thesis, Pontifícia Universidade Católica do Rio de Janeiro, Rio de Janeiro, Brazil, 2010.
- [7] A. de Oliveira, R. Sampaio-Neto, and C. A. Medina, "Blind detection techniques for IEEE 802.15.4a communication system," in *Wireless Communication Systems (ISWCS), 2012 International Symposium on*. Aug. 2012, pp. 1064–1068, IEEE, doi:10.1109/ISWCS.2012.6328531.
- [8] A. de Oliveira, R. Sampaio-Neto, and C. A. Medina, "Detecção de sinais em sistemas UWB baseados no padrão IEEE 802.15.4a," in *XXIX Simpósio Brasileiro de Telecomunicações*, Curitiba, Brazil, 2011.
- [9] F. A. Sandoval and R. Sampaio-Neto, "Novos receptores com posto reduzido e sua aplicação em sistemas DS-CDMA," in *XXXI Simpósio Brasileiro de Telecomunicações*, Fortaleza, Brazil, 2013, doi:10.14209/sbrt.2013.2.
- [10] F. A. Sandoval, "Novos receptores com posto reduzido e suas aplicações em sistemas baseados em DS-CDMA," Master thesis, Pontifícia Universidade Católica do Rio de Janeiro, Rio de Janeiro, Brazil, 2013.
- [11] H. L. VanTrees, *Optimum Array Processing*, Wiley Interscience, 2002.
- [12] O. L. Frost III, "An algorithm for linearly constrained adaptive array processing," *Proc. IEEE*, vol. 60, no. 18, pp. 926–935, Aug. 1972, doi:10.1109/PROC.1972.8817.
- [13] J. A. Apolinario, M. L. R. De Campos, and C. P. Bernal O, "The constrained conjugate gradient algorithm," *IEEE Signal Process. Lett.*, vol. 7, no. 12, pp. 351–354, Dec. 2000, doi:10.1109/97.883366.
- [14] J. Ward, "Space-time adaptive processing," Technical Report 1015, MIT Lincoln Laboratory, Lexington, MA, 1994.
- [15] H. Hotelling, *Analysis of a Complex of Statistical Variables into Principal Components*, Baltimore, MD: Warwick and York, 1933.
- [16] A. Haimovich, "The eigencanceler: adaptive radar by eigenanalysis methods," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 32, no. 2, pp. 532–542, Apr. 1996, doi:10.1109/7.489498.
- [17] J. S. Goldstein and I. S. Reed, "Theory of partially adaptive radar," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 33, no. 4, pp. 1309–1325, Oct. 1997, doi:10.1109/7.625132.
- [18] C. Lanczos, "Solution of systems of linear equations by minimized iterations," *Journal of Research of the National Bureau of Standards*, vol. 49, no. 1, pp. 33–53, Jul. 1952.
- [19] M. Joham and M. D. Zoltowski, "Interpretation of the multistage nested wiener filter in the Krylov subspace framework," Technical Report TR-ECE-00-51/TUM-LNS-TR-00-6, Purdue University/Munich University of Technology, [Online]. Available: http://cobweb.ecn.purdue.edu/~Lijmikiedz/research/msnwf_tutorial.pdf, 2000.
- [20] G. H. Golub and C. F. Van Loan, *Matrix Computations*, The Johns Hopkins Univ. Press, second edition, 1984.
- [21] W. E. Arnoldi, "The principle of minimized iterations in the solution of the matrix eigenvalue problem," *Quarterly of Applied Mathematics*, vol. 9, no. 1, pp. 17–29, Jan. 1951, doi:10.1090/qam/42792.
- [22] J. S. Goldstein, I. S. Reed, and L. L. Scharf, "A multistage representation of the wiener filter based on orthogonal projections," *IEEE Trans. Inf. Theory*, vol. 44, no. 7, pp. 2943–2959, Nov. 1998, doi:10.1109/18.737524.
- [23] G. Dietl, M. D. Zoltowski, and M. Joham, "Recursive reduced-rank adaptive equalization for wireless communications," *Proc. of SPIE*, vol. 4395, pp. 16–27, Apr. 2001, doi:10.1117/12.438286.
- [24] D. A. Pados and G. N. Karystinos, "An iterative algorithm for the computation of the MVDR filter," *IEEE Trans. Signal Process.*, vol. 49, no. 2, pp. 290–300, Feb. 2001, doi:10.1109/78.902111.
- [25] B. L. Mathews, L. Mili, and A. I. Zaghloul, "Auxiliary vector selection algorithms for adaptive beamforming," in *2005 IEEE Antennas and Propagation Society International Symposium*, Jul. 2005, vol. 3A, pp. 271–274 vol. 3A, doi:10.1109/APS.2005.1552233.
- [26] R. C. de Lamare and R. Sampaio-Neto, "Reduced-rank adaptive filtering based on joint iterative optimization of adaptive filters," *IEEE Signal Process. Lett.*, vol. 14, no. 12, pp. 980–983, Dec. 2007, doi:10.1109/LSP.2007.907995.
- [27] L. Wang and R. C. DeLamare, "Low-complexity constrained adaptive reduced-rank beamforming algorithms," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 49, no. 4, pp. 2114–2128, Oct. 2013, doi:10.1109/TAES.2013.6621805.
- [28] H. Cox, R. M. Zeeskind, and M. M. Owen, "Robust adaptive beamforming," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. 35, no. 10, pp. 1365–1376, Oct. 1987, doi:10.1109/TASSP.1987.1165054.
- [29] M. H. Er and A. Cantoni, "Derivative constraints for broad-band element space antenna array processors," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. 31, no. 6, pp. 1378–1393, Dec. 1983, doi:10.1109/TASSP.1983.1164219.
- [30] K. M. Buckley and L. J. Griffiths, "An adaptive generalized sidelobe canceller with derivative constraints," *IEEE Trans. Antennas Propag.*, vol. 34, no. 3, pp. 311–319, Mar. 1986, doi:10.1109/TAP.1986.1143832.
- [31] R. Couillet, F. Pascal, and J. W. Silverstein, "A joint robust estimation and random matrix framework with application to array processing," in *2013 IEEE International Conference on Acoustics, Speech and Signal Processing*, May 2013, pp. 6561–6565, doi:10.1109/ICASSP.2013.6638930.
- [32] D. D. Feldman and L. J. Griffiths, "A projection approach for robust adaptive beamforming," *IEEE Trans. Signal Process.*, vol. 42, no. 4, pp. 867–876, Apr. 1994, doi:10.1109/78.285650.
- [33] A. B. Gershman, N. D. Sidiropoulos, S. Shahbazpahari, M. Bengtsson, and B. Ottersten, "Convex optimization-based beamforming," *IEEE Signal Process. Mag.*, vol. 27, no. 3, pp. 62–75, May 2010, doi:10.1109/MSP.2010.936015.
- [34] S. A. Vorobyov, A. B. Gershman, and Zhi-Quan Luo, "Robust adaptive beamforming using worst-case performance optimization: a solution to the signal mismatch problem," *IEEE Trans. Signal Process.*, vol. 51, no. 2, pp. 313–324, Feb. 2003, doi: 10.1109/TSP.2002.806865.



Aline de Oliveira received the Diploma degree in telecommunications engineering from Universidade Federal Fluminense (UFF), Brazil, in 2007, and the M.Sc. and D.Sc. degrees, both in electrical engineering, from Pontifícia Universidade Católica do Rio de Janeiro (PUC-Rio), Brazil, in 2010 and 2017, respectively. Since 2010, she has been a researcher at Instituto de Pesquisas da Marinha (IPqM), Brazil.

Her research interests are radar signal processing, STAP and signal processing for electronic warfare and wideband digital receivers.



José Mauro Fortes was born in Rio de Janeiro, Brazil, in 1950. He received the B.S. and M.Sc. degrees in electrical engineering (telecommunications) from the Pontifícia Universidade Católica do Rio de Janeiro (PUC-Rio), in 1973 and 1976, respectively, and the M.Sc. and Ph.D. degrees in electrical engineering from Stanford University, USA, in 1978 and 1980, respectively. He returned to PUC-Rio, in June 1980, and was a Professor with the Electrical Engineering Department, until 2018, working at the University Center for Telecommunications Studies

(CETUC), where he was the Head of the Communication Systems Group, from 2000 to 2018. He is currently an Independent Consultant in satellite communications. In 1992, while on sabbatical leave, he was a Researcher with the General Electric Research and Development Center, Schenectady, USA. He has published several articles in national and international journals and conferences. He has coordinated a number of research projects and has been a Consultant in satellite communications for several private companies and telecommunications agencies.

His main research interests include communication theory, satellite communications, estimation theory, and digital transmission. For two terms (from 1996 to 2000), he was the President of the Brazilian Telecommunications Society (an IEEE ComSoc Sister Society), and for 13 years, he was the Vice-Chairman of the ITU-R Study Group 4 (Fixed Satellite Service).



Raimundo Sampaio-Neto received the Diploma and the M.Sc. degrees from Pontifícia Universidade Católica do Rio de Janeiro (PUC-Rio), Brazil, in 1975 and 1978, respectively, and the Ph.D. degree from the University of Southern California (USC), Los Angeles, in 1983, all in electrical engineering. From November 1983 to June 1984 he was a Post-Doctoral fellow at the Communication Sciences Institute of the Department of Electrical Engineering at USC, and a member of the technical staff of Axiomatic Corporation, Los Angeles. He is currently

a researcher with the Center for Studies in Telecommunications (CETUC) and an Associate Professor of the Department of Electrical Engineering, PUC-Rio, where he has been since July 1984. In 1991, he was a Visiting Professor in the Department of Electrical Engineering at USC. He has participated in various projects and consulted for several private companies and government agencies.

His areas of interest include communication theory, digital transmission and signal processing for communications, areas in which he has published more than 200 papers in referred journals and conferences. He was a co-organizer of the Session on Recent Results for the IEEE Workshop on Information Theory, 1992, Salvador, Brazil. He also served as Technical Program Cochairman for IEEE Global Telecommunications Conference (Globecom'99) held in Rio de Janeiro and as a Technical Program Member for several national and international conferences. He was in office for three terms on the Board of Directors of the Brazilian Communications Society (SBrT). He served as a member of its Advisory Council for four terms and as Associate Editor of the society journal: Journal of Communication and Information Systems.

He is currently an Emerit Member of SBrT.



Fabian David Backx received the Diploma degree in telecommunications engineering from Universidade Federal Fluminense (UFF), Brazil, in 2002, and the M.Sc. and D.Sc. degrees, both in electrical engineering, from Pontifícia Universidade Católica do Rio de Janeiro (PUC-Rio), Brazil, in 2004 and 2009, respectively. Since 2010, he has been a researcher at Instituto de Pesquisas da Marinha (IPqM), Brazil.

His areas of interest include statistical signal processing, communication theory, digital transmission and signal processing for communications, radar and

electronic warfare.

He is a member of the IEEE.