Reduced-Dimension Robust Capon Beamforming Using Krylov-Subspace Techniques

Samuel D. Somasundaram*, *Member, IEEE*, Peng Li, *Member, IEEE*,
Nigel H. Parsons *Member, IEEE*, and Rodrigo C. de Lamare *Senior Member, IEEE*

Abstract

We present low-complexity, quickly converging robust adaptive beamformers, for beamforming large arrays in snapshot deficient scenarios. The proposed algorithms are derived by combining data-dependent Krylov-subspace based dimensionality reduction, using the Powers-of-R or Conjugate Gradient techniques, with ellipsoidal uncertainty set based robust Capon beamformer methods. Further, we provide a detailed computational complexity analysis and consider the efficient implementation of automatic, online dimension-selection rules. We illustrate the benefits of the proposed approaches using simulated data.

Index Terms

Reduced-dimension robust Capon beamforming, robust adaptive beamforming, reduced-rank adaptive beamforming, Krylov subspace, powers-of-R, conjugate gradient, low sample support, large arrays.

This work was supported in part by MOD under contract from the Centre for Defence Enterprise, CDE 26553 - Contract No. DSTLX1000070981.

S. D. Somasundaram and N. H. Parsons are with the General Sonar Studies Group, Thales U.K., Poseidon House, Ashurst Drive, Bird Hall Lane, Cheadle Heath, Stockport, Cheshire, SK3 0XB, U.K. (email: sam.somasundaram@uk.thalesgroup.com, sdsomasundaram@hotmail.com; nigel_parsons@tiscali.co.uk).

R. C. de Lamare and P. Li are with the Communications Research Group, Department of Electronics, University of York, Heslington, North Yorkshire, York YO10 5DD, U.K. (email: rcdl500@ohm.york.ac.uk; peng.li@york.ac.uk).

I. Introduction

In this work, we are interested in designing low-complexity, robust adaptive beamformers for beamforming large arrays in snapshot deficient scenarios. Such beamformers are applicable to radar and passive sonar systems, in which large aperture, many-element arrays that operate in highly dynamic environments are often encountered and where the number of snapshots K that are available for weight estimation is significantly less than the number of array elements M [1], [2]. The challenges are to a) provide robustness to signal-of-interest (SOI) array steering vector (ASV) modeling errors, which arise from, e.g., calibration and pointing errors and can cause severe SOI cancellation, b) ensure adequate convergence on the limited data available, and c) ensure a sufficiently low complexity to permit real-time implementation on the often limited computational resources that are available.

Many robust adaptive techniques have been proposed to alleviate the deleterious effects of ASV mismatch, including using multiple linear constraints [3]–[6], diagonal loading [7], eigenspace projection [8], and more recently, the robust Capon beamformers (RCBs), which exploit ellipsoidal, including spherical, uncertainty sets of the ASV [9]–[13], and their extensions [2], [14]–[21]. Due to their widespread popularity and the fact that they have addressed the mismatch problem in several application areas (see, e.g., [22]–[29]), herein, we focus on using RCB techniques for alleviating the effects of ASV mismatch. Other notable recent methods for addressing the mismatch problem can be found in [30]–[33]. Whilst the RCB (under spherical uncertainty) converges significantly faster than, e.g., the minimum variance distortionless response (MVDR) beamformer, there is still often both a need for and potential for improvement, especially when $K < M$. Further, the $\mathcal{O}(M^3)$ complexity eigenvalue decomposition required to compute the full-dimension RCB is prohibitive for real-time implementation on the large arrays considered here.

Reduced-dimension (or reduced-rank) methods, which project the data onto a low-rank subspace, are often used to speed-up the convergence of adaptive beamformers and reduce their computational complexity. Common data-independent methods include subarray pre-processing [34]–[37], full-aperture beamspace [38]–[40], and discrete prolate spheroidal sequences [40], [41]. The *ad hoc* nature of the data-independent methods and the need to improve performance, motivated research into statistically optimum data-dependent methods, the first of which were the eigenbasis based principal components [42] and cross-spectral metric [1], [43], [44] methods. Whilst these could provide huge reductions in sample support over conventional full-dimension methods [45], they were soon superseded by several algorithms that are related to each other as they operate in the same Krylov-subspace [46]–[54], but differ in

the way that they expand the subspace and then form the adaptive weights. These Krylov-subspace algorithms include the multistage Wiener filter [1], [55], the auxiliary vector method [56], [57], the conjugate gradient technique [48], [58]–[62], and the reduced-rank MVDR algorithms in [63], [64]. Due to the popularity and the proven performance of these Krylov-subspace techniques in several application areas (see, e.g., [1], [46], [47], [49], [51], [52], [56], [59], [61], [63], [65]–[67]), we here focus on exploiting Krylov-subspace dimensionality reduction, but remark that there are other methods available (see, e.g. [68], [69]). Significant drawbacks of the aforementioned Krylov-subspace methods are that they are sensitive to SOI ASV errors and to the selected rank.

The main contribution of this paper is combining Krylov-subspace dimensionality reduction and robust Capon beamforming, to produce a new family of Krylov-subspace based reduced-dimension robust Capon beamformers (Krylov-RDRCBs). By exploiting Krylov-subspace methods, we obtain faster convergence and lower complexity than the RCB. By exploiting RCB methods, we address the sensitivity of the Krylov-based methods to SOI ASV errors and rank over-determination. We examine four existing Krylov-subspace expansion methods for computing dimensionality reduction transforms, which have different degrees of numerical stability and associated computational costs. They are the non-orthogonal Power-of-R (NO-PoR) method, previously used to implement the multistage Wiener filter in [52], the method used in [63], [64], which we here term the orthogonal PoR (O-PoR) method, and two methods based on the conjugate gradient (CG) technique [58], where one exploits the CG direction vectors and we term the CG method, and where the other exploits the orthogonal CG residuals and we term the orthogonal CG (O-CG) method. To form the Krylov-RDRCBs, we use the RDRCB framework proposed in [2], [70]. Here, we extend the framework, by providing new results that allow faster computation of the reduced-dimension ellipsoids that are needed after rank reduction and that are useful when exploiting *any* type of data-dependent dimensionality reduction transform (and not just the Krylov-subspace methods considered here). We term the new Krylov-RDRCBs the NO-PoR-RDRCB, the O-PoR-RDRCB, the CG-RDRCB, and the O-CG-RDRCB, which we derive assuming a known dimension and for both spherical and non-degenerate ellipsoidal uncertainty. We also exploit the fact that the conjugate-gradient direction vectors diagonalize the reduced-dimension covariance to derive an efficient implementation of the CG-RDRCB. We provide a detailed analysis of the computational complexities of the Krylov-RDRCBs. Since, in practice, the optimal dimension (or rank) is often unknown *a priori*, we consider how to efficiently implement the automatic rank selection rules proposed in [52], [71] with each of the Krylov subspace-expansion methods. Further, we propose two extremely low-complexity stopping rules for the CG-RDRCBs, which stop subspace expansion either when the condition number of the

reduced-dimension covariance exceeds a threshold or when breakdown of the conjugate orthogonality of the direction vectors is detected. Extensive simulations are used to assess the new Krylov-RDRCBs, with and without automatic rank selection, and in comparison to existing Krylov-subspace based reduced-dimension MVDRs and full dimension delay-and-sum, MVDR, and RCBs, for beamforming large planar arrays in snapshot deficient scenarios. To summarize, the contributions of this work[1] comprise 1) the new family of Krylov-RDRCBs, 2) the extension of the RDRCB framework for data-dependent dimensionality reduction, 3) the efficient implementation of rank-selection rules, 4) a detailed computational complexity analysis and 5) a simulation study for beamforming large arrays in snapshot deficient scenarios.

The remainder of this paper is organized as follows. In Section II, after discussing the notation and operation counting conventions (for evaluating computational complexity) used throughout the paper, we summarize the RCB approach and the Krylov-subspace methods used for rank reduction. In Section III, we review the RDRCB framework and then present new results for exploiting data-dependent dimensionality reduction. In Section IV, we exploit these results to form the Krylov-RDRCBs. We derive a computationally efficient implementation of the CG-RDRCB in Section IV-A. In Section V, we analyze the complexity of the proposed Krylov-RDRCBs for a given rank N. In Section VI, we incorporate rank selection rules. In Section VII, on simulated data, we illustrate the benefits of the proposed algorithms for beamforming large arrays in snapshot deficient scenarios. Finally, we draw our conclusions in Section VIII.

II. Background

Here, we summarize the notation and operation counting conventions used throughout the paper, as well as the RCB and Krylov-subspace expansion techniques.

A. Notation and Operation Counting Conventions

In the following, $E\{\cdot\}$, $(\cdot)^T$, $(\cdot)^H$, $(\cdot)^{-1}$ and $(\cdot)^\dagger$ denote the expectation, transpose, Hermitian transpose, inverse and Moore-Penrose pseudo-inverse operators, respectively. Furthermore, $\mathbb{C}^{M\times N}$, $\mathbf{I}_N$, $\|\cdot\|_2$, $\mathcal{R}(\mathbf{X})$, $\mathcal{N}(\mathbf{X})$, $\mathbf{N}_{\mathbf{X}}^l$, $\mathbf{\Pi}_{\mathbf{X}}$ and $\mathbf{\Pi}_{\mathbf{X}}^\perp$ denote the space of $M \times N$ complex matrices, the $N \times N$ identity matrix, the two-norm, the column-space of $\mathbf{X}$, the null-space of $\mathbf{X}$, a basis for the left null-space of $\mathbf{X}$, the orthogonal projector onto the column space of $\mathbf{X}$ and the orthogonal projector onto the space perpendicular to the column space of $\mathbf{X}$, respectively. Moreover, $\mathbf{X} \geq 0$ or $\mathbf{X} > 0$ mean that the Hermitian symmetric matrix $\mathbf{X}$ is positive semi-definite or positive definite, respectively. We define $\mathbf{X}^{\frac{1}{2}}$ such that $\mathbf{X} = \mathbf{X}^{\frac{H}{2}}\mathbf{X}^{\frac{1}{2}}$, where $\mathbf{X}^{\frac{H}{2}} = (\mathbf{X}^{\frac{1}{2}})^H$.

[1]Part of this work was presented at ICASSP 2013 [72].

An operation is defined as one complex multiplication plus addition (and is approximately equivalent to four real multiplications and additions) such that, for $\mathbf{X} \in \mathbb{C}^{M\times N}$ and $\mathbf{Y} \in \mathbb{C}^{N\times P}$, $\mathbf{XY}$ requires MNP operations. If $\mathbf{X} \in \mathbb{C}^{M\times M}$ is a diagonal matrix and $\mathbf{Y} \in \mathbb{C}^{M\times P}$ is a general matrix, then $\mathbf{XY}$ requires MP operations. Computing the eigenvalue decomposition (EVD) or the inverse of an $M \times M$ Hermitian symmetric matrix is assumed to require $\mathcal{O}(4\frac{1}{3}M^3)$ or $\mathcal{O}(M^3)$ operations[2], respectively [73].

B. Robust Capon Beamforming

The kth full-dimension (or element-space) array snapshot $\mathbf{x}_k \in \mathbb{C}^M$ is modeled as

$$\mathbf{x}_k = \mathbf{a}_0 s_{0,k} + \mathbf{n}_k, \tag{1}$$

where $\mathbf{a}_0$, $s_{0,k}$ and $\mathbf{n}_k$ denote the true SOI ASV, the SOI complex amplitude and an additive zero-mean complex Gaussian vector that incorporates the noise and the interference. Assuming $s_{0,k}$ is zero mean and uncorrelated with $\mathbf{n}_k$, the array covariance matrix can be written as $\mathbf{R_x} = E\left\{\mathbf{x}_k\mathbf{x}_k^H\right\} = \sigma_0^2\mathbf{a}_0\mathbf{a}_0^H + \mathbf{Q_x}$, where $\mathbf{R_x} > 0$, $\sigma_0^2 = E\left\{|s_{0,k}|^2\right\}$ is the SOI power, and $\mathbf{Q_x} = E\left\{\mathbf{n}_k\mathbf{n}_k^H\right\}$. In practice, $\mathbf{R_x}$ is often replaced by the sample covariance matrix estimate

$$\hat{\mathbf{R}}_\mathbf{x} = \frac{1}{K}\sum_{k=1}^{K}\mathbf{x}_k\mathbf{x}_k^H, \tag{2}$$

formed from K snapshots. In [10], [11] (see, also [13]), RCBs were derived by solving $\max_{\sigma^2,\mathbf{a}}\ \sigma^2$ s.t. $\mathbf{R_x} - \sigma^2\mathbf{a}\mathbf{a}^H \geq 0,\ \mathbf{a} \in \mathcal{E}_M(\bar{\mathbf{a}}, \mathbf{E})$, which can be reduced to [10], [11]

$$\min_{\mathbf{a}} \mathbf{a}^H\mathbf{R}_\mathbf{x}^{-1}\mathbf{a} \quad \text{s.t.} \quad \mathbf{a} \in \mathcal{E}_M(\bar{\mathbf{a}}, \mathbf{E}). \tag{3}$$

The M-dimensional, full-dimension (or element-space) ellipsoid $\mathcal{E}_M(\bar{\mathbf{a}}, \mathbf{E})$ is parameterized by its center $\bar{\mathbf{a}}$, often representing the assumed ASV, and by its principal semi-axes, given by the unit-norm left singular vectors of $\mathbf{E}^{-\frac{1}{2}}$, where $\mathbf{E} \geq 0 \in \mathbb{C}^{M\times M}$, scaled by the corresponding singular values, and can be written as

$$\mathcal{E}_M(\bar{\mathbf{a}}, \mathbf{E}) = \left\{\mathbf{a} \in \mathbb{C}^M \;\middle|\; [\mathbf{a} - \bar{\mathbf{a}}]^H\mathbf{E}[\mathbf{a} - \bar{\mathbf{a}}] \leq 1\right\}. \tag{4}$$

[2]In the limit of large M, for an M-dimensional real symmetric matrix, $4/3M^3$, approx. $30M^2$ and $3M^3$ real operations are needed to reduce to tridiagonal form (via Householder reduction), diagonalize the tridiagonal matrix (to obtain the eigenvalues) and obtain the eigenvectors, respectively. Ignoring M^2 terms, in the limit of large M, $4\frac{1}{3}M^3$ real operations are needed to compute the EVD of a real symmetric matrix. The eigen-pairs of a complex M dimensional Hermitian matrix can be obtained from the eigen-pairs of a $2M$ dimensional real symmetric matrix called the augmented matrix. Using this method would require $4\frac{1}{3}(2M)^3 = 34\frac{2}{3}M^3$ real operations. However, in principle, an algorithm designed specifically for complex matrices could give a further factor of 2 reduction in complexity over the augmented matrix method, leading to $17\frac{1}{3}M^3$ real operations. We here assume that dividing by 4 gives the number of complex operations needed to compute the EVD of a Hermitian matrix as $4\frac{1}{3}M^3$.

To avoid the trivial solution $\mathbf{a} = \mathbf{0}$ of (3), it is necessary to assume that

$$\bar{\mathbf{a}}^H \mathbf{E} \bar{\mathbf{a}} > 1. \tag{5}$$

When $\mathbf{E} = \epsilon^{-1}\mathbf{I}$, (4) reduces to a spherical uncertainty set, $\|\mathbf{a} - \bar{\mathbf{a}}\|_2^2 \leq \epsilon$, with radius $\sqrt{\epsilon}$ and (5) becomes $\|\bar{\mathbf{a}}\|_2^2 > \epsilon$. Whilst conservative spherical uncertainty sets are most often used in practice, more general ellipsoids can provide a more accurate description of ASV uncertainty and yield better performance in certain scenarios [13], [15], [74]. For non-degenerate ellipsoids with $\mathbf{E} > 0$, we can factor $\mathbf{E} = \mathbf{E}^{\frac{H}{2}}\mathbf{E}^{\frac{1}{2}}$ and form $\breve{\mathbf{a}} = \mathbf{E}^{\frac{1}{2}}\mathbf{a}$, $\breve{\bar{\mathbf{a}}} = \mathbf{E}^{\frac{1}{2}}\bar{\mathbf{a}}$ and $\breve{\mathbf{R}}_{\mathbf{x}} = \mathbf{E}^{\frac{1}{2}}\mathbf{R}_{\mathbf{x}}\mathbf{E}^{\frac{H}{2}}$. Then, (3) can be re-written using the following spherical constraint [11]

$$\min_{\breve{\mathbf{a}}} \breve{\mathbf{a}}^H \breve{\mathbf{R}}_{\mathbf{x}}^{-1} \breve{\mathbf{a}} \text{ s.t. } \left\|\breve{\mathbf{a}} - \breve{\bar{\mathbf{a}}}\right\|_2^2 \leq 1. \tag{6}$$

As shown in [11], (6) can be solved using Lagrange multipliers, via the EVD $\breve{\mathbf{R}}_{\mathbf{x}} = \breve{\mathbf{U}}\breve{\mathbf{\Lambda}}\breve{\mathbf{U}}^H$, where $\breve{\mathbf{\Lambda}}$ is a diagonal matrix containing the eigenvalues $\breve{\lambda}_1 \geq \breve{\lambda}_2 \ldots \geq \breve{\lambda}_M$ in non-increasing order on its main diagonal and where $\breve{\mathbf{U}}$ contains the associated eigenvectors. Computing $\breve{\mathbf{z}} = \breve{\mathbf{U}}^H \breve{\mathbf{a}}$ in M^2 operations and letting $\breve{z}_m$ denote the mth entry of $\breve{\mathbf{z}}$, the Lagrange multiplier μ is found by solving $f(\mu) = \sum_{m=1}^{M} \frac{|\breve{z}_m|^2}{(1+\mu\breve{\lambda}_m)^2} - 1 = 0$ via Newton line search, which updates μ according to $\mu_{k+1} = \mu_k - \frac{f(\mu_k)}{f'(\mu_k)}$, where μ_k is the value of μ at the kth iteration and $f'(\mu) = -2\sum_{m=1}^{M} \frac{|\breve{z}_m|^2 \breve{\lambda}_m}{(1+\mu\breve{\lambda}_m)^3}$. The search terminates when $f(\mu_{k+1})$ is sufficiently close to zero. Tight bounds on μ, which can be used to initialize the search, are given in [11]. Each Newton iteration takes around $6M$ operations. Once μ has been found, the solution to (6) is formed as $\hat{\breve{\mathbf{a}}} = \breve{\mathbf{a}} - \breve{\mathbf{U}}(\mathbf{I} + \mu\breve{\mathbf{\Lambda}})^{-1}\breve{\mathbf{z}}$, which requires an additional $\mathcal{O}(M[M+2])$ operations. Thus, solving (6) for $\hat{\breve{\mathbf{a}}}$ once the EVD of $\breve{\mathbf{R}}_{\mathbf{x}}$ has been computed, requires $O(2M[M+1] + n_{iter}6M)$ operations, where n_{iter} denotes the number of iterations in the Newton search. The solution to (3) is formed as $\hat{\mathbf{a}}_{0,\text{RCB}} = \mathbf{E}^{-\frac{1}{2}}\hat{\breve{\mathbf{a}}}$. The RCB power estimate is formed as $\hat{\sigma}^2_{0,\text{RCB}} = \frac{\|\hat{\mathbf{a}}_{0,\text{RCB}}\|_2^2/M}{\hat{\mathbf{a}}^H_{0,\text{RCB}}\mathbf{R}_{\mathbf{x}}^{-1}\hat{\mathbf{a}}_{0,\text{RCB}}}$ and the weight vector as $\hat{\mathbf{w}}_{\text{RCB}} = \frac{\mathbf{R}_{\mathbf{x}}^{-1}\hat{\mathbf{a}}_{0,\text{RCB}}}{\hat{\mathbf{a}}^H_{0,\text{RCB}}\mathbf{R}_{\mathbf{x}}^{-1}\hat{\mathbf{a}}_{0,\text{RCB}}}$. The operation counts for computing the full-dimension RCB are summarized in Table I.

TABLE I

ONLINE OPERATION COUNTS FOR RCB.

	Spherical	Non-Degenerate
$\breve{\mathbf{R}}_{\mathbf{x}} = \breve{\mathbf{U}}\breve{\mathbf{\Lambda}}\breve{\mathbf{U}}^H$	$\mathcal{O}(4\frac{1}{3}M^3)$	
Solve for $\hat{\breve{\mathbf{a}}}$	$\mathcal{O}(2M[M+1] + n_{\text{iter}}6M)$	
$\hat{\mathbf{a}}_0 = \mathbf{E}^{-\frac{1}{2}}\hat{\breve{\mathbf{a}}}$	0	$\mathcal{O}(M^2)$
$\hat{\mathbf{w}}_{\text{RCB}}$	$\mathcal{O}(M[M+1])$	

C. Krylov-Subspace Bases

Here, we summarize the PoR and CG methods for obtaining rank-N Krylov-subspace bases or dimension reducing transforms, which span the following N-dimensional Krylov subspace

$$\mathcal{K}_N = \mathcal{R}\left(\left[\begin{array}{cccc} \bar{\mathbf{a}} & \hat{\mathbf{R}}_{\mathbf{x}}\bar{\mathbf{a}} & \ldots \hat{\mathbf{R}}_{\mathbf{x}}^{N-1}\bar{\mathbf{a}} \end{array}\right]\right). \tag{7}$$

1) Non-Orthogonal Powers-of-R: The standard non-orthogonal PoR (NO-PoR) basis is given by [52]

$$\mathbf{D} = \left[\begin{array}{cccc} \frac{\bar{\mathbf{a}}}{\|\bar{\mathbf{a}}\|_2} & \frac{\hat{\mathbf{R}}_{\mathbf{x}}\bar{\mathbf{a}}}{\left\|\hat{\mathbf{R}}_{\mathbf{x}}\bar{\mathbf{a}}\right\|_2} & \cdots & \frac{\hat{\mathbf{R}}_{\mathbf{x}}^{N-1}\bar{\mathbf{a}}}{\left\|\hat{\mathbf{R}}_{\mathbf{x}}^{N-1}\bar{\mathbf{a}}\right\|_2} \end{array}\right], \tag{8}$$

which can be formed iteratively, by starting with $\boldsymbol{\kappa}_1 = \bar{\mathbf{a}}$ and $\mathbf{d}_1 = \frac{\bar{\mathbf{a}}}{\|\bar{\mathbf{a}}\|_2}$, and calculating, for $i = 2, \ldots, N$,

$$\boldsymbol{\kappa}_i = \hat{\mathbf{R}}_{\mathbf{x}}\boldsymbol{\kappa}_{i-1} \tag{9}$$

and $\mathbf{d}_i = \frac{\boldsymbol{\kappa}_i}{\|\boldsymbol{\kappa}_i\|_2}$. Calculating $\boldsymbol{\kappa}_i$ from $\boldsymbol{\kappa}_{i-1}$ uses $\mathcal{O}(M^2)$ operations and then calculating $\mathbf{d}_i$ uses $\mathcal{O}(M)$ operations. Thus, calculating the rank-N NO-PoR transform uses $\mathcal{O}(NM[M+1])$ operations.

2) Orthogonal Powers-of-R: In [63], an alternative PoR-based transform was suggested for applications in which the model order is highly variable and time-varying, and can be formed similarly to the NO-PoR transform, by replacing (9) with $\boldsymbol{\kappa}_i = \boldsymbol{\Pi}^{\perp}_{\mathbf{D}_{i-1}}\hat{\mathbf{R}}_{\mathbf{x}}\boldsymbol{\kappa}_{i-1}$, where $\boldsymbol{\Pi}^{\perp}_{\mathbf{D}_{i-1}} = \mathbf{I} - \sum_{k=1}^{i-1}\mathbf{d}_k\mathbf{d}_k^H$. We term the resulting transform the orthogonal PoR (O-PoR) transform as its columns are orthogonal. To form a new column $\mathbf{d}_i$ efficiently, first compute $\tilde{\boldsymbol{\kappa}}_i = \hat{\mathbf{R}}_{\mathbf{x}}\boldsymbol{\kappa}_{i-1}$ in $\mathcal{O}(M^2)$ operations, then $\boldsymbol{\kappa}_i = \tilde{\boldsymbol{\kappa}}_i - \mathbf{D}_{i-1}\mathbf{D}_{i-1}^H\tilde{\boldsymbol{\kappa}}_i$, where $\mathbf{D}_{i-1} = \left[\begin{array}{ccc} \mathbf{d}_1 & \ldots & \mathbf{d}_{i-1} \end{array}\right]$, in a further $\mathcal{O}(2(i-1)M + M)$ operations and then calculate $\mathbf{d}_i$ from $\boldsymbol{\kappa}_i$ in a further $\mathcal{O}(M)$ operations. Thus, computing the N-dimensional O-PoR transform requires $\mathcal{O}(NM[M+1+N])$ operations.

3) Conjugate Gradient Method: The CG algorithm minimizes the function $f(\mathbf{w}) = \mathbf{w}^H\hat{\mathbf{R}}_{\mathbf{x}}\mathbf{w} - \mathbf{w}^H\bar{\mathbf{a}} - \bar{\mathbf{a}}^H\mathbf{w} + constant$ along a set of $\hat{\mathbf{R}}_{\mathbf{x}}$-orthogonal search directions and thus, iteratively solves $\hat{\mathbf{R}}_{\mathbf{x}}\mathbf{w} = \bar{\mathbf{a}}$ [48], [58], [75]. Starting with $\mathbf{w}_1 = \mathbf{0}$, $\mathbf{d}_1 = \bar{\mathbf{a}}$ and $\mathbf{r}_1 = -\bar{\mathbf{a}}$, for $i = 2, \ldots, N$, calculate [48]

$$\begin{aligned} \alpha_i &= -\frac{\mathbf{d}_i^H\mathbf{r}_i}{\mathbf{d}_i^H\hat{\mathbf{R}}_{\mathbf{x}}\mathbf{d}_i} \\ \mathbf{w}_{i+1} &= \mathbf{w}_i + \alpha_i\mathbf{d}_i \\ \mathbf{r}_{i+1} &= \mathbf{r}_i + \alpha_i\hat{\mathbf{R}}_{\mathbf{x}}\mathbf{d}_i \\ \beta_i &= \frac{\mathbf{d}_i^H\hat{\mathbf{R}}_{\mathbf{x}}\mathbf{r}_{i+1}}{\mathbf{d}_i^H\hat{\mathbf{R}}_{\mathbf{x}}\mathbf{d}_i} \\ \mathbf{d}_{i+1} &= -\mathbf{r}_{i+1} + \beta_i\mathbf{d}_i, \end{aligned} \tag{10}$$

where $\mathbf{d}_i$ and $\mathbf{r}_i$ denote the ith conjugate direction vector and residual vector, respectively. The direction vectors are formed by conjugation of the residuals and therefore, at the ith iteration, they both span the

same subspace, denoted $\mathcal{D}_i = \mathcal{R}([\ \mathbf{r}_1 \quad \ldots \quad \mathbf{r}_i\ ]) = \mathcal{R}([\ \mathbf{d}_1 \quad \ldots \quad \mathbf{d}_i\ ])$. From (10), we see that $\mathbf{r}_{i+1}$ is a linear combination of $\mathbf{r}_i$ and $\hat{\mathbf{R}}_{\mathbf{x}}\mathbf{d}_i$. Since $\mathbf{d}_i \in \mathcal{D}_i$, the new subspace $\mathcal{D}_{i+1}$ is formed from the union of $\mathcal{D}_i$ and $\hat{\mathbf{R}}_{\mathbf{x}}\mathcal{D}_i$, so that [75]

$$\begin{aligned} \mathcal{D}_i &= \mathcal{R}([\ \mathbf{d}_1 \quad \hat{\mathbf{R}}_{\mathbf{x}}\mathbf{d}_1 \quad \ldots \quad \hat{\mathbf{R}}_{\mathbf{x}}^{i-1}\mathbf{d}_1\ ]) \\ &= \mathcal{R}([\ \mathbf{r}_1 \quad \hat{\mathbf{R}}_{\mathbf{x}}\mathbf{r}_1 \quad \ldots \quad \hat{\mathbf{R}}_{\mathbf{x}}^{i-1}\mathbf{r}_1\ ]). \end{aligned} \tag{11}$$

Thus, with $\mathbf{d}_1 = \bar{\mathbf{a}}$ and $\mathbf{r}_1 = -\bar{\mathbf{a}}$ in (11), it clear that $\mathcal{D}_N$, which is the subspace spanned by the first N residuals or direction vectors, is equivalent to the Krylov subspace in (7). We term the dimension reducing transform, $\mathbf{D} = [\mathbf{d}_1 \ldots \mathbf{d}_N]$, formed using the direction vectors the CG transform, whilst we term $\mathbf{D} = \left[\frac{\mathbf{r}_1}{\|\mathbf{r}_1\|_2} \ldots \frac{\mathbf{r}_N}{\|\mathbf{r}_N\|_2}\right]$ formed using the residuals the orthogonal CG (O-CG) transform. The cost of computing $\hat{\mathbf{R}}_{\mathbf{x}}\mathbf{d}_i$ is $\mathcal{O}(M^2)$. Given $\hat{\mathbf{R}}_{\mathbf{x}}\mathbf{d}_i$, the cost of computing α_i is $\mathcal{O}(2M)$. Updating $\mathbf{r}_{i+1}$ is $\mathcal{O}(M)$. The cost of computing β_i, given $\hat{\mathbf{R}}_{\mathbf{x}}\mathbf{d}_i$ and the denominator of α_i is $\mathcal{O}(M)$. Then, updating $\mathbf{d}_{i+1}$ is $\mathcal{O}(M)$. Note that we do not need to calculate updates to $\mathbf{w}$ to find $\mathbf{D}$. Thus, the cost of computing a new column of the CG transform is $\mathcal{O}(M^2 + 5M)$. For the O-CG transform, an extra $\mathcal{O}(M)$ operations are needed to compute the norm $\|\mathbf{r}_{i+1}\|_2$. Thus, the total operations needed to calculate the CG and O-CG transforms are $\mathcal{O}(NM[M+5])$ and $\mathcal{O}(NM[M+6])$, respectively. Table II summarizes the operations needed to compute each of the Krylov-subspace dimension reducing transforms (DRTs).

III. Data-Dependent Reduced-Dimension Robust Capon Beamforming Framework

Here, we first summarize the existing reduced-dimension robust Capon beamforming (RDRCB) framework, proposed in [2], [70] for combining any form of dimensionality reduction with robust Capon beamforming, and then extend it for data-dependent dimensionality reduction.

TABLE II

Operations for computing rank-N Krylov-subspace DRTs.

Method	Operations
NO-PoR	$\mathcal{O}(NM[M+1])$
O-PoR	$\mathcal{O}(NM[M+1+N])$
CG	$\mathcal{O}(NM[M+5])$
O-CG	$\mathcal{O}(NM[M+6])$

A. Summary of Existing RDRCB Framework

In reduced-dimension methods, the kth full-dimension snapshot, $\mathbf{x}_k \in \mathbb{C}^M$, is projected onto an N-dimensional subspace (with $N < M$) using a dimension reducing transformation $\mathbf{D} \in \mathbb{C}^{M\times N}$, yielding the reduced-dimension snapshot, $\mathbf{y}_k = \mathbf{D}^H\mathbf{x}_k$, where $\mathbf{y}_k \in \mathbb{C}^N$. Using (1), the reduced-dimension snapshot can be modeled as $\mathbf{y}_k = \mathbf{b}_0 s_{0,k} + \tilde{\mathbf{n}}_k$, with $\mathbf{b}_0 = \mathbf{D}^H\mathbf{a}_0$ and $\tilde{\mathbf{n}}_k = \mathbf{D}^H\mathbf{n}_k$, giving a reduced-dimension covariance $\mathbf{R}_\mathbf{y} = E\left\{\mathbf{y}_k\mathbf{y}_k^H\right\} = \sigma_0^2\mathbf{b}_0\mathbf{b}_0^H + \mathbf{Q}_\mathbf{y}$, with $\mathbf{Q}_\mathbf{y} = E\left\{\tilde{\mathbf{n}}_k\tilde{\mathbf{n}}_k^H\right\} = \mathbf{D}^H\mathbf{Q}_\mathbf{x}\mathbf{D}$. Since $\mathbf{R}_\mathbf{y}$ is unavailable *a priori*, it is often replaced with $\hat{\mathbf{R}}_\mathbf{y} = \frac{1}{K}\sum_{k=1}^K \mathbf{y}_k\mathbf{y}_k^H = \mathbf{D}^H\hat{\mathbf{R}}_\mathbf{x}\mathbf{D}$, with $\hat{\mathbf{R}}_\mathbf{x}$ given in (2). This leads to the following RDRCB optimization problem $\max_{\sigma^2,\mathbf{b}}\ \sigma^2$ s.t. $\mathbf{R}_\mathbf{y} - \sigma^2\mathbf{b}\mathbf{b}^H \geq 0,\ \mathbf{b} \in \mathcal{E}_N(\bar{\mathbf{b}}, \mathbf{F})$, where $\mathbf{b} = \mathbf{D}^H\mathbf{a}$ and $\mathcal{E}_N(\bar{\mathbf{b}}, \mathbf{F})$ is a reduced-dimension ellipsoid, yielding [2], [70]

$$\min_{\mathbf{b}} \mathbf{b}^H\mathbf{R}_\mathbf{y}^{-1}\mathbf{b} \quad \text{s.t.} \quad \mathbf{b} \in \mathcal{E}_N(\bar{\mathbf{b}}, \mathbf{F}). \tag{12}$$

The following propagation theorem is used to remove components from the full-dimension ellipsoid, $\mathcal{E}_M(\bar{\mathbf{a}}, \mathbf{E})$, that belong to $\mathcal{N}(\mathbf{D}^H)$, to produce a reduced-dimension ellipsoid.

Theorem 1: [2], [70] The propagation of the full-dimension ellipsoid $\mathcal{E}_M(\bar{\mathbf{a}}, \mathbf{E})$ (4), where $\mathbf{E} \geq 0 \in \mathbb{C}^{M\times M}$, through the mapping defined by $\mathbf{D}^H\mathbf{a} - \mathbf{I}_N\mathbf{b} = \mathbf{0}$, where $\mathbf{D} \in \mathbb{C}^{M\times N}$ has full column rank, yields the ellipsoid $\mathcal{E}_N(\bar{\mathbf{b}}, \mathbf{F})$ [see ellipsoid definition in (4)] with

$$\bar{\mathbf{b}} = \mathbf{D}^H\bar{\mathbf{a}} \tag{13}$$

$$\mathbf{F} = \mathbf{D}^\dagger(\mathbf{E} - \mathbf{E}\mathbf{N}_\mathbf{D}^l[(\mathbf{N}_\mathbf{D}^l)^H\mathbf{E}\mathbf{N}_\mathbf{D}^l]^\dagger(\mathbf{N}_\mathbf{D}^l)^H\mathbf{E})(\mathbf{D}^\dagger)^H. \tag{14}$$

When the full-dimension set is a sphere, so that in (4) $\mathbf{E} = \epsilon^{-1}\mathbf{I}$, the simpler expression $\mathbf{F} = \epsilon^{-1}(\mathbf{D}^H\mathbf{D})^{-1}$ is obtained, which reduces to $\mathbf{F} = \epsilon^{-1}\mathbf{I}_N$ if $\mathbf{D}$ has orthogonal columns.

Given, $\mathcal{E}_N(\bar{\mathbf{b}}, \mathbf{F})$ and $\mathbf{R}_\mathbf{y}$, an estimate $\hat{\mathbf{b}}_0$ is found by solving (12) using standard RCB results [see Section II-B]. Given $\hat{\mathbf{b}}_0$, the RDRCB weight vector is formed as

$$\hat{\mathbf{w}}_{\text{RDRCB}} = \frac{\mathbf{R}_\mathbf{y}^{-1}\hat{\mathbf{b}}_0}{\hat{\mathbf{b}}_0^H\mathbf{R}_\mathbf{y}^{-1}\hat{\mathbf{b}}_0}, \tag{15}$$

which operates on the reduced-dimension data. The weights $\hat{\mathbf{w}}_{\text{RDRCB,ES}} = \mathbf{D}\hat{\mathbf{w}}_{\text{RDRCB}}$ operate on the original full-dimension data. An estimate of $\mathbf{a}_0$ is formed as $\hat{\mathbf{a}}_0 = (\mathbf{D}^H)^\dagger\hat{\mathbf{b}}_0 = \mathbf{D}(\mathbf{D}^H\mathbf{D})^{-1}\hat{\mathbf{b}}_0$, illustrating that $\hat{\mathbf{a}}_0 \in \mathcal{R}(\mathbf{D})$. Given $\hat{\mathbf{a}}_0$, the RDRCB SOI power estimate is formed as

$$\hat{\sigma}^2_{0,\text{RDRCB}} = \frac{(\|\hat{\mathbf{a}}_0\|_2^2/M)}{\hat{\mathbf{b}}_0^H\mathbf{R}_\mathbf{y}^{-1}\hat{\mathbf{b}}_0} = \frac{\hat{\mathbf{b}}_0^H(\mathbf{D}^H\mathbf{D})^{-1}\hat{\mathbf{b}}_0}{M\hat{\mathbf{b}}_0^H\mathbf{R}_\mathbf{y}^{-1}\hat{\mathbf{b}}_0}. \tag{16}$$

B. Extension for Data-Dependent Dimensionality Reduction

Here, we provide new results, useful for exploiting data-dependent dimensionality reduction. Examining (14), we observe that, in general, $\mathbf{D}^{\dagger}$, $\mathbf{N}_{\mathbf{D}}^{l}$ and $[(\mathbf{N}_{\mathbf{D}}^{l})^{H}\mathbf{E}\mathbf{N}_{\mathbf{D}}^{l}]^{\dagger}$ need computing to obtain $\mathbf{F}$. The operations required to compute $\mathbf{N}_{\mathbf{D}}^{l}$ and $[(\mathbf{N}_{\mathbf{D}}^{l})^{H}\mathbf{E}\mathbf{N}_{\mathbf{D}}^{l}]^{\dagger}$ are $\mathcal{O}([M-N]^3)$ complexity. Moreover, when $N << M$, which is of interest to us here, the complexity $\mathcal{O}([M-N]^3) \approx \mathcal{O}(M^3)$. For data-independent dimensionality reduction that uses fixed $\mathbf{D}$, which was the only type of dimensionality reduction considered in [2], [70], these terms can be computed off-line, so that their computation does not affect the online implementation of the RDRCB. However, when exploiting data-dependent dimensionality reduction, such as the Krylov-subspace methods considered in this paper, these terms need computing online every time $\mathbf{D}$ is updated and therefore, the $\mathcal{O}([M-N]^3) \approx \mathcal{O}(M^3)$ complexity needed to compute $\mathbf{F}$ using (14) is prohibitive.

To address this problem, here, we note that if the original full-dimension ellipsoid is non-degenerate, such that $\mathbf{E} > 0$, we can beneficially re-write the expression for $\mathbf{F}$ in (14). Firstly, we remark that the assumption of non-degenerate ellipsoids, which only excludes flat ellipsoids, is not prohibitive in practice, since some flexibility in all dimensions is actually beneficial for robustness to, typically present, arbitrary ASV errors [74]. For $\mathbf{E} > 0$, we can write $[(\mathbf{N}_{\mathbf{D}}^{l})^{H}\mathbf{E}\mathbf{N}_{\mathbf{D}}^{l}]^{\dagger} = [(\mathbf{N}_{\mathbf{D}}^{l})^{H}\mathbf{E}\mathbf{N}_{\mathbf{D}}^{l}]^{-1}$, which we insert into (14) to give

$$\begin{aligned}
\mathbf{F} &= \mathbf{D}^{\dagger}(\mathbf{E} - \mathbf{E}\mathbf{N}_{\mathbf{D}}^{l}[(\mathbf{N}_{\mathbf{D}}^{l})^{H}\mathbf{E}\mathbf{N}_{\mathbf{D}}^{l}]^{-1}(\mathbf{N}_{\mathbf{D}}^{l})^{H}\mathbf{E})(\mathbf{D}^{\dagger})^{H} \\
&= \mathbf{D}^{\dagger}\mathbf{E}^{\frac{1}{2}}\mathbf{\Pi}^{\perp}_{\mathbf{E}^{\frac{1}{2}}\mathbf{N}_{\mathbf{D}}^{l}}\mathbf{E}^{\frac{1}{2}}(\mathbf{D}^{\dagger})^{H} \\
&= \mathbf{D}^{\dagger}\mathbf{E}^{\frac{1}{2}}\mathbf{\Pi}_{\mathbf{E}^{-\frac{1}{2}}\mathbf{D}}\mathbf{E}^{\frac{1}{2}}(\mathbf{D}^{\dagger})^{H} \\
&= \mathbf{D}^{\dagger}\mathbf{D}\left[\mathbf{D}^{H}\mathbf{E}^{-1}\mathbf{D}\right]^{-1}\mathbf{D}^{H}\left(\mathbf{D}^{\dagger}\right)^{H} \\
&= \left[\mathbf{D}^{H}\mathbf{E}^{-1}\mathbf{D}\right]^{-1}, \qquad (17)
\end{aligned}$$

with $\mathbf{\Pi}^{\perp}_{\mathbf{E}^{\frac{1}{2}}\mathbf{N}_{\mathbf{D}}^{l}} = \mathbf{I} - \mathbf{E}^{\frac{1}{2}}\mathbf{N}_{\mathbf{D}}^{l}[(\mathbf{N}_{\mathbf{D}}^{l})^{H}\mathbf{E}\mathbf{N}_{\mathbf{D}}^{l}]^{-1}(\mathbf{N}_{\mathbf{D}}^{l})^{H}\mathbf{E}^{\frac{1}{2}}$ and $\mathbf{\Pi}_{\mathbf{E}^{-\frac{1}{2}}\mathbf{D}} = \mathbf{E}^{-\frac{1}{2}}\mathbf{D}[\mathbf{D}^{H}\mathbf{E}^{-1}\mathbf{D}]^{-1}\mathbf{D}^{H}\mathbf{E}^{-\frac{1}{2}}$. To obtain the third equality in (17), we note that $\mathbf{\Pi}^{\perp}_{\mathbf{E}^{\frac{1}{2}}\mathbf{N}_{\mathbf{D}}^{l}}$ projects onto $\mathcal{N}([\mathbf{E}^{\frac{1}{2}}\mathbf{N}_{\mathbf{D}}^{l}]^{H})$ and that the columns of $\mathbf{E}^{-\frac{1}{2}}\mathbf{D}$ form a basis for $\mathcal{N}([\mathbf{E}^{\frac{1}{2}}\mathbf{N}_{\mathbf{D}}^{l}]^{H})$, that is $(\mathbf{N}_{\mathbf{D}}^{l})^{H}\mathbf{E}^{\frac{1}{2}}\mathbf{E}^{-\frac{1}{2}}\mathbf{D} = (\mathbf{N}_{\mathbf{D}}^{l})^{H}\mathbf{D} = \mathbf{0}$, such that $\mathbf{\Pi}_{\mathbf{E}^{-\frac{1}{2}}\mathbf{D}} \equiv \mathbf{\Pi}^{\perp}_{\mathbf{E}^{\frac{1}{2}}\mathbf{N}_{\mathbf{D}}^{l}}$. Since $\mathbf{E}^{-1}$ can be computed off-line, the online computation of $\mathbf{F}$ reduces to the $\mathcal{O}(NM[M+N])$ complexity needed to multiply the factors in $\mathbf{F}^{-1} = \left[\mathbf{D}^{H}\mathbf{E}^{-1}\mathbf{D}\right]$ and then the complexity required to compute the $N \times N$ inverse. However, for a general non-degenerate ellipsoid, the factors $\mathbf{F}^{\frac{1}{2}}$, $\mathbf{F}^{\frac{H}{2}}$ and $\mathbf{F}^{-\frac{1}{2}}$ need computing [see Section II-B], which can all be obtained from the EVD of $\mathbf{F}^{-1} = \left[\mathbf{D}^{H}\mathbf{E}^{-1}\mathbf{D}\right]$, so that the $N \times N$ inverse never needs computing. Thus, for large M compared

to N, the contribution here is reducing the approximately $O(M^3)$ online complexity that would be required using (14) to approximately $\mathcal{O}(NM^2)$. As the EVD of $\breve{\mathbf{R}}_{\mathbf{y}} = \mathbf{F}^{\frac{1}{2}}\mathbf{R}_{\mathbf{y}}\mathbf{F}^{\frac{H}{2}}$ is also needed, two N-dimensional EVDs will be required for general non-degenerate ellipsoidal uncertainty sets. As noted earlier, when the full-dimension set is a sphere and $\mathbf{D}$ is orthogonal, $\mathbf{F} = \epsilon^{-1}\mathbf{I}_N$, so that only one EVD is required,

The summary steps for data-dependent reduced-dimension robust Capon beamforming, where it is assumed that $\mathbf{E}^{-1}$ has been pre-computed off-line, are

1) Evaluate the dimension reducing transform (DRT) $\mathbf{D}$ (using, e.g., one of the methods described in Section II-C).
2) Calculate $\mathbf{R}_{\mathbf{y}} = \mathbf{D}^H\mathbf{R}_{\mathbf{x}}\mathbf{D}$ and $\bar{\mathbf{b}} = \mathbf{D}^H\bar{\mathbf{a}}$.
3) For a spherical set and orthogonal $\mathbf{D}$, skip to step 5.
   For a spherical set and non-orthogonal $\mathbf{D}$, compute the EVD of $\epsilon\left[\mathbf{D}^H\mathbf{D}\right] = \mathbf{V}\mathbf{\Gamma}\mathbf{V}^H$.
   For a non-degenerate set, compute the EVD of $\left[\mathbf{D}^H\mathbf{E}^{-1}\mathbf{D}\right] = \mathbf{V}\mathbf{\Gamma}\mathbf{V}^H$.
4) Form $\mathbf{F}^{\frac{1}{2}} = \mathbf{\Gamma}^{-\frac{1}{2}}\mathbf{V}^H$, $\mathbf{F}^{\frac{H}{2}} = \mathbf{V}\mathbf{\Gamma}^{-\frac{1}{2}}$ and $\mathbf{F}^{-\frac{1}{2}} = \mathbf{V}\mathbf{\Gamma}^{\frac{1}{2}}$. Then, compute $\breve{\mathbf{R}}_{\mathbf{y}} = \mathbf{F}^{\frac{1}{2}}\mathbf{R}_{\mathbf{y}}\mathbf{F}^{\frac{H}{2}}$ and $\breve{\bar{\mathbf{b}}} = \mathbf{F}^{\frac{1}{2}}\bar{\mathbf{b}}$.
5) Solve $\min_{\breve{\mathbf{b}}} \breve{\mathbf{b}}^H\breve{\mathbf{R}}_{\mathbf{y}}^{-1}\breve{\mathbf{b}}$ s.t. $\left\|\breve{\mathbf{b}} - \breve{\bar{\mathbf{b}}}\right\|_2^2 \leq 1$ to obtain $\hat{\breve{\mathbf{b}}}$. For the case of a spherical set with squared radius ϵ and an orthogonal $\mathbf{D}$, we set $\breve{\mathbf{R}}_{\mathbf{y}} = \epsilon^{-1}\mathbf{R}_{\mathbf{y}}$ and $\breve{\bar{\mathbf{b}}} = \epsilon^{-\frac{1}{2}}\bar{\mathbf{b}}$.
6) Given $\hat{\breve{\mathbf{b}}}$, we form $\hat{\mathbf{b}}_0 = \mathbf{F}^{-\frac{1}{2}}\hat{\breve{\mathbf{b}}}$ and form the weights using (15) and the power using (16). We remark that $\hat{\breve{\mathbf{w}}}_{\text{RDRCB}} = \frac{\breve{\mathbf{R}}_{\mathbf{y}}^{-1}\hat{\breve{\mathbf{b}}}}{\hat{\breve{\mathbf{b}}}^H\breve{\mathbf{R}}_{\mathbf{y}}^{-1}\hat{\breve{\mathbf{b}}}} = \mathbf{F}^{-\frac{H}{2}}\hat{\mathbf{w}}_{\text{RDRCB}}$.

Note that, for spherical uncertainty sets, as $[\mathbf{D}^H\mathbf{D}]^{-1} = \epsilon\mathbf{V}\mathbf{\Gamma}^{-1}\mathbf{V}^H$, we can write the numerator of the power term (16) efficiently as $\hat{\mathbf{b}}_0^H(\mathbf{D}^H\mathbf{D})^{-1}\hat{\mathbf{b}}_0 = \epsilon\left\|\hat{\breve{\mathbf{b}}}\right\|_2^2$.

IV. Krylov-Subspace Reduced-Dimension Robust Capon Beamforming

Here, we describe how to form four different Krylov-RDRCBs. We use the framework described in Section III, which is general and can be used with any form of dimensionality reduction. Krylov-RDRCBs can be formed by using a Krylov dimensionality reduction transform $\mathbf{D}$, obtained using one of the Krylov-subspace expansion techniques described in Section II-C, with steps 1-6 in Section III-B.

When the NO-PoR transform (8) is used, we term the resulting algorithm the NO-PoR-RDRCB. Since the columns of the NO-PoR transform are not orthogonal, following the discussion in Section III-B, computing the NO-PoR-RDRCB weights requires two N-dimensional EVDs, even for spherical full-dimension sets.

When the O-PoR transform, described in Section II-C2, is used, we term the resulting algorithm the O-PoR-RDRCB. Since the O-PoR transform has orthogonal columns, only one EVD is required to compute the O-PoR-RDRCB for spherical uncertainty sets. For non-degenerate uncertainty sets, two EVDs are required.

When the dimension reducing transform is formed from the conjugate-gradient direction vectors, we term the resulting algorithm the CG-RDRCB. Since the direction vectors are not orthogonal, following the discussion in Section III-B, we would expect that two EVDs would be needed to compute the CG RDRCB. However, below in Section IV-A, we illustrate how a fast CG-RDRCB can be obtained by exploiting that the direction vectors diagonalize the sample covariance matrix.

When the dimension reducing transform is formed from the conjugate gradient residuals, we term the resulting algorithm the O-CG-RDRCB, which requires a single EVD under spherical uncertainty and two under non-degenerate uncertainty.

A. Fast Conjugate Gradient-Based RDRCB

Here, we illustrate how only one N-dimensional EVD is required to obtain the CG-RDRCB under either spherical or non-degenerate uncertainty. The key is noting that when $\mathbf{D}$ is formed using the CG algorithm,

$$\hat{\mathbf{R}}_{\mathbf{y}} = \mathbf{D}^H \hat{\mathbf{R}}_{\mathbf{x}} \mathbf{D} = \mathbf{\Lambda}_{\text{CG}}, \tag{18}$$

where $\mathbf{\Lambda}_{\text{CG}}$ is a diagonal matrix given by

$$\mathbf{\Lambda}_{\text{CG}} = \text{diag}\left\{ \left[\begin{array}{ccc} \mathbf{d}_1^H \hat{\mathbf{R}}_{\mathbf{x}} \mathbf{d}_1 & \ldots & \mathbf{d}_N^H \hat{\mathbf{R}}_{\mathbf{x}} \mathbf{d}_N \end{array} \right] \right\}. \tag{19}$$

We remark that $\mathbf{d}_i^H \hat{\mathbf{R}}_{\mathbf{x}} \mathbf{d}_i$ has already been calculated for $i = 1, \ldots, N-1$ when calculating α_i in (10). Therefore, evaluation of (18) reduces to the $\mathcal{O}(M[M+1])$ operations needed to compute $\mathbf{d}_N^H \hat{\mathbf{R}}_{\mathbf{x}} \mathbf{d}_N$.

In general, we will be solving

$$\min_{\mathbf{b}} \mathbf{b}^H \mathbf{R}_{\mathbf{y}}^{-1} \mathbf{b} \text{ s.t. } \left[\mathbf{b} - \bar{\mathbf{b}}\right]^H \mathbf{F} \left[\mathbf{b} - \bar{\mathbf{b}}\right] \leq 1. \tag{20}$$

Usually, at this stage, one would transform with $\mathbf{F}^{\frac{1}{2}}$ to give a spherical uncertainty set. However, from (18), we observe that $\mathbf{R}_{\mathbf{y}}^{-1} = \mathbf{\Lambda}_{\text{CG}}^{-1}$, so that (20) can be written as

$$\min_{\mathbf{b}} \mathbf{b}^H \mathbf{\Lambda}_{\text{CG}}^{-1} \mathbf{b} \text{ s.t. } \left[\mathbf{b} - \bar{\mathbf{b}}\right]^H \mathbf{F} \left[\mathbf{b} - \bar{\mathbf{b}}\right] \leq 1. \tag{21}$$

Letting $\check{\mathbf{b}} = \mathbf{\Lambda}_{\text{CG}}^{-\frac{1}{2}} \mathbf{b}$, $\check{\bar{\mathbf{b}}} = \mathbf{\Lambda}_{\text{CG}}^{-\frac{1}{2}} \bar{\mathbf{b}}$ and $\mathbf{M} = \mathbf{\Lambda}_{\text{CG}}^{-\frac{1}{2}} \mathbf{D}^H \mathbf{E}^{-1} \mathbf{D} \mathbf{\Lambda}_{\text{CG}}^{-\frac{H}{2}}$, we can rewrite (21) as

$$\min_{\check{\mathbf{b}}} \check{\mathbf{b}}^H \check{\mathbf{b}} \text{ s.t. } \left[\check{\mathbf{b}} - \check{\bar{\mathbf{b}}}\right]^H \mathbf{M}^{-1} \left[\check{\mathbf{b}} - \check{\bar{\mathbf{b}}}\right] \leq 1. \tag{22}$$

The associated Lagrangian is formed as

$$L(\check{\mathbf{b}},\mu)=\check{\mathbf{b}}^H\check{\mathbf{b}}+\mu\left(\left[\check{\mathbf{b}}-\check{\bar{\mathbf{b}}}\right]^H\mathbf{M}^{-1}\left[\check{\mathbf{b}}-\check{\bar{\mathbf{b}}}\right]-1\right), \tag{23}$$

where μ is a real-valued Lagrange multiplier. Setting $\frac{\partial L(\check{\mathbf{b}},\mu)}{\partial\check{\mathbf{b}}^H}=\mathbf{0}$ yields

$$\begin{aligned}\hat{\check{\mathbf{b}}} &= \mu\left(\mathbf{I}+\mu\mathbf{M}^{-1}\right)^{-1}\mathbf{M}^{-1}\check{\bar{\mathbf{b}}}=\left(\frac{\mathbf{M}}{\mu}+\mathbf{I}\right)^{-1}\check{\bar{\mathbf{b}}}\\ &= \check{\bar{\mathbf{b}}}-\left[\mu\mathbf{M}^{-1}+\mathbf{I}\right]^{-1}\check{\bar{\mathbf{b}}}.\end{aligned} \tag{24}$$

Using (24) in the constraint equation yields

$$\begin{aligned}h(\hat{\check{\mathbf{b}}},\mu) &= \left[\hat{\check{\mathbf{b}}}-\check{\bar{\mathbf{b}}}\right]^H\mathbf{M}^{-1}\left[\hat{\check{\mathbf{b}}}-\check{\bar{\mathbf{b}}}\right]\\ &= \check{\bar{\mathbf{b}}}^H\left[\mu\mathbf{M}^{-1}+\mathbf{I}\right]^{-1}\mathbf{M}^{-1}\left[\mu\mathbf{M}^{-1}+\mathbf{I}\right]^{-1}\check{\bar{\mathbf{b}}}^H.\end{aligned} \tag{25}$$

Letting $\mathbf{M}=\mathbf{U}\mathbf{\Lambda}\mathbf{U}^H$ denote the EVD of $\mathbf{M}$, where $\mathbf{\Lambda}=\text{diag}\left\{\left[\begin{array}{ccc}\lambda_1 & \ldots & \lambda_N\end{array}\right]\right\}$ is a diagonal matrix containing the eigenvalues in non-increasing order on its main diagonal and $\mathbf{U}$ contains the associated eigenvectors, we can write (25) as

$$\begin{aligned}h(\hat{\check{\mathbf{b}}},\mu) &= \check{\bar{\mathbf{b}}}^H\mathbf{U}\left(\mu\mathbf{\Lambda}^{-1}+\mathbf{I}\right)^{-1}\mathbf{\Lambda}^{-1}\left(\mu\mathbf{\Lambda}^{-1}+\mathbf{I}\right)^{-1}\mathbf{U}^H\check{\bar{\mathbf{b}}}\\ &= \sum_{n=1}^{N}\frac{\lambda_n|c_n|^2}{\left(\mu+\lambda_n\right)^2},\end{aligned} \tag{26}$$

where c_n is the nth element of $\mathbf{c}=\mathbf{U}^H\check{\bar{\mathbf{b}}}$. Since we can write $\mathbf{M}=\mathbf{M}^{\frac{1}{2}}\mathbf{M}^{\frac{H}{2}}$, where $\mathbf{M}^{\frac{1}{2}}=\mathbf{\Lambda}_{\text{CG}}^{-\frac{1}{2}}\mathbf{D}^H\mathbf{E}^{-\frac{1}{2}}$, we know that $\mathbf{M}$ is non-negative definite [76], [77] and therefore, it has non-negative real eigenvalues. Thus, $h(\hat{\check{\mathbf{b}}},\mu)$ is a monotonically decreasing function of $\mu>0$. A trivial solution is inevitable at $\mu=0$, which gives

$$h(\hat{\check{\mathbf{b}}},0)=\check{\bar{\mathbf{b}}}^H\mathbf{M}^{-1}\check{\bar{\mathbf{b}}}=\bar{\mathbf{b}}^H\left[\mathbf{D}^H\mathbf{E}^{-1}\mathbf{D}\right]^{-1}\bar{\mathbf{b}}=\bar{\mathbf{b}}^H\mathbf{F}\bar{\mathbf{b}}. \tag{27}$$

Thus, to avoid a trivial solution, we require that $\bar{\mathbf{b}}^H\mathbf{F}\bar{\mathbf{b}}>1$, which, if satisfied, means that the solution to $h(\hat{\check{\mathbf{b}}},\mu)=1$ occurs at $\mu>0$. Further, as $\lim_{\mu\to\infty}h(\hat{\check{\mathbf{b}}},\mu)=0$, there is a unique solution $\mu>0$ to $h(\hat{\check{\mathbf{b}}},\mu)=1$ or equivalently to $\bar{h}(\hat{\check{\mathbf{b}}},\mu)=h(\hat{\check{\mathbf{b}}},\mu)-1=0$, which can be found, e.g., by Newton search. The Newton search iterates $\mu_{k+1}=\mu_k-\frac{\bar{h}(\hat{\check{\mathbf{b}}},\mu_k)}{\bar{h}'(\hat{\check{\mathbf{b}}},\mu_k)}$, where $h'(\hat{\check{\mathbf{b}}},\mu)=-2\sum_{n=1}^{N}\frac{\lambda_n|c_n|^2}{(\mu+\lambda_n)^3}$, until $\bar{h}(\hat{\check{\mathbf{b}}},\mu_{k+1})$ is sufficiently close to zero. Each iteration of our Newton search requires $\mathcal{O}(6N)$ operations. Once the EVD of $\mathbf{M}$ has been calculated, $\mathcal{O}(2N[N+1]+\tilde{n}_{iter}6N)$ operations are required to solve for $\hat{\check{\mathbf{b}}}$; N^2 to compute $\mathbf{c}$, $\tilde{n}_{iter}6N$ to compute the Lagrange multiplier and a further N^2+2N operations to compute $\hat{\check{\mathbf{b}}}=\check{\bar{\mathbf{b}}}-\mathbf{U}\left[\mu\mathbf{\Lambda}^{-1}+\mathbf{I}\right]^{-1}\mathbf{c}$. The solution to (21) is formed as $\hat{\mathbf{b}}_0=\mathbf{\Lambda}_{\text{CG}}^{\frac{1}{2}}\hat{\check{\mathbf{b}}}$. We can use $\hat{\mathbf{b}}_0$ and $\mathbf{R}_{\mathbf{y}}^{-1}=\mathbf{\Lambda}_{\text{CG}}^{-1}$ in (15) to form the adaptive weights and form the power estimate using (16).

The main steps for computing the CG-RDRCB weights are as follows

1) Evaluate the CG transform $\mathbf{D}$ via (10) and $\mathbf{\Lambda}_{\text{CG}}$ using (19).
2) Form $\breve{\mathbf{b}} = \mathbf{\Lambda}_{\text{CG}}^{-\frac{1}{2}}\mathbf{D}^H\bar{\mathbf{a}}$ and $\mathbf{M} = \mathbf{\Lambda}_{\text{CG}}^{-\frac{1}{2}}\mathbf{D}^H\mathbf{E}^{-1}\mathbf{D}\mathbf{\Lambda}_{\text{CG}}^{-\frac{H}{2}}$.
3) Calculate the EVD, $\mathbf{U}\mathbf{\Lambda}\mathbf{U}^H = \mathbf{M}$.
4) Form $\mathbf{c} = \mathbf{U}^H\breve{\mathbf{b}}$ and find $\mu > 0$ from (26) via Newton search.
5) Form $\hat{\breve{\mathbf{b}}} = \breve{\mathbf{b}} - \mathbf{U}\left[\mu\mathbf{\Lambda}^{-1} + \mathbf{I}\right]^{-1}\mathbf{c}$.
6) Form $\hat{\mathbf{b}}_0 = \mathbf{\Lambda}_{\text{CG}}^{\frac{1}{2}}\hat{\breve{\mathbf{b}}}$ and $\mathbf{w}_{\text{RDRCB}} = \frac{\mathbf{\Lambda}_{\text{CG}}^{-1}\hat{\mathbf{b}}_0}{\hat{\mathbf{b}}_0^H\mathbf{\Lambda}_{\text{CG}}^{-1}\hat{\mathbf{b}}_0} = \frac{\mathbf{\Lambda}_{\text{CG}}^{-\frac{1}{2}}\hat{\breve{\mathbf{b}}}}{\left\|\hat{\breve{\mathbf{b}}}\right\|_2^2}$.

For spherical uncertainty sets, note that $[\mathbf{D}^H\mathbf{D}]^{-1} = \epsilon\mathbf{\Lambda}_{\text{CG}}^{-\frac{H}{2}}\mathbf{M}^{-1}\mathbf{\Lambda}_{\text{CG}}^{-\frac{1}{2}} = \epsilon\mathbf{\Lambda}_{\text{CG}}^{-\frac{H}{2}}\mathbf{U}\mathbf{\Lambda}^{-1}\mathbf{U}^H\mathbf{\Lambda}_{\text{CG}}^{-\frac{1}{2}}$, so that the numerator in (16) can be efficiently computed using $\hat{\mathbf{b}}_0^H[\mathbf{D}^H\mathbf{D}]^{-1}\hat{\mathbf{b}}_0 = \epsilon\left\|\mathbf{\Lambda}^{-\frac{1}{2}}\mathbf{U}^H\hat{\breve{\mathbf{b}}}\right\|_2^2 = \epsilon\left\|\mathbf{\Lambda}^{-\frac{1}{2}}\left[\mathbf{c} - \left[\mu\mathbf{\Lambda}^{-1} + \mathbf{I}\right]^{-1}\mathbf{c}\right]\right\|_2^2$ with only an additional $\mathcal{O}(3N)$ computations. For a (general) non-degenerate ellipsoid, we will need to compute $\left[\mathbf{D}^H\mathbf{D}\right]^{-1}$.

V. Complexity Analysis

Here, we analyze the complexity of the proposed Krylov-RDRCBs for an assumed rank N. The operation counts summarized in Tables III and IV relate to Steps 1–6 in Section III-B for either the NO-PoR-RDRCB or the O-PoR-RDRCB under either spherical or non-degenerate uncertainty. In Step 5,

TABLE III

Online operation counts for spherical PoR-based RDRCBs.

	NO-PoR	O-PoR
Evaluate DRT	$\mathcal{O}(NM[M+1])$	$\mathcal{O}(NM[M+1+N])$
$\mathbf{R_y} = \mathbf{D}^H\mathbf{R_x}\mathbf{D}$	$\mathcal{O}(NM[M+N])$	
$\bar{\mathbf{b}} = \mathbf{D}^H\bar{\mathbf{a}}$	$\mathcal{O}(NM)$	
$\left(\mathbf{D}^H\mathbf{D}\right)$	$\mathcal{O}(MN^2)$	0
$\mathbf{V}\mathbf{\Gamma}\mathbf{V}^H = \mathbf{D}^H\mathbf{D}$	$\mathcal{O}(4\frac{1}{3}N^3)$	0
$\mathbf{F}^{\frac{1}{2}} = \mathbf{\Gamma}^{-\frac{1}{2}}\mathbf{V}^H$	$\mathcal{O}(N^2)$	0
$\breve{\bar{\mathbf{b}}} = \mathbf{F}^{\frac{1}{2}}\bar{\mathbf{b}}$	$\mathcal{O}(N^2)$	0
$\breve{\mathbf{R}}_{\mathbf{y}} = \mathbf{F}^{\frac{1}{2}}\mathbf{R_y}\mathbf{F}^{\frac{H}{2}}$	$\mathcal{O}(2N^3)$	0
EVD($\breve{\mathbf{R}}_{\mathbf{y}}$)	$\mathcal{O}(4\frac{1}{3}N^3)$	
Solve for $\hat{\breve{\mathbf{b}}}$	$\mathcal{O}(2N[N+1] + n_{\text{iter}}6N)$	
$\hat{\mathbf{b}}_0 = \mathbf{F}^{-\frac{1}{2}}\hat{\breve{\mathbf{b}}} = \mathbf{V}\mathbf{\Gamma}^{\frac{1}{2}}\hat{\breve{\mathbf{b}}}$	$\mathcal{O}(N[N+1])$	0
$\mathbf{w}_{\text{RDRCB}}$	$\mathcal{O}(N[N+1])$	
Power	0	

we need to solve a reduced-dimension RCB optimization for $\hat{\hat{\mathbf{b}}}$. This can be solved using the standard RCB techniques described in Section II-B so that once the EVD of $\breve{\mathbf{R}}_{\mathbf{y}}$ is calculated, $\hat{\hat{\mathbf{b}}}$ is found via Newton search in $\mathcal{O}(2N[N+1]+n_{\text{iter}}6N)$ operations. The operation counts for the O-CG-RDRCB are the same as those for the O-PoR-RDRCB, except for the count evaluating the dimensionality reduction transform. The operations in Table V relate to Steps 1–6 in Section IV-A for the efficient implementation of the CG-RDRCB under either spherical or non-degenerate uncertainty. In Step 4, $\tilde{n}_{iter}$ iterations are needed to find the Lagrange multiplier. Empirically, we have observed that there is little difference between the values of n_{iter} and $\tilde{n}_{iter}$ and in the following calculations set them both equal to 15. When non-degenerate sets are exploited in the NO-PoR or CG RDRCBs, significant savings in complexity can be achieved if power-estimation scaling is not required, as then $\left[\mathbf{D}^H\mathbf{D}\right]^{-1}$ does not need computing, which would require a further $\mathcal{O}(N^2[M+N])$ operations. Fig. 1 shows the relative complexities of the different algorithms as N is increased from 1 to M, for $M = 320$, where for the NO-PoR and CG RDRCBs under non-degenerate (ND) uncertainty, we distinguish between applications that only require weight estimation and those that also need power estimation. The results show that the efficient CG-based algorithms are the least complex.

It is interesting to consider how the complexities of the proposed Krylov-RDRCBs compare to the non-robust Krylov-subspace based reduced-dimension MVDRs (Krylov-RDMVDRs), which can be formed by using (15) with $\bar{\mathbf{b}}$ instead of $\hat{\mathbf{b}}_0$, and to the full-dimension RCB whose complexity is summarized in Table I. For the 320 element array examined in the numerical examples and when exploiting spherical full-dimension uncertainty sets, the CG-RDRCB with N = 5 is over 200 times less complex than using an RCB. In surveillance applications, such as passive sonar, often many hundreds of beams are formed to span the angular space in both azimuth and elevation. The complexity of the RCB is dominated by the $\mathcal{O}(4\frac{1}{3}M^3)$ operations needed to compute the EVD of $\breve{\mathbf{R}}_{\mathbf{x}}$ [see Section II-B]. For spherical uncertainty, $\breve{\mathbf{R}}_{\mathbf{x}} = \epsilon^{-1}\mathbf{R}_{\mathbf{x}}$, so that the EVD of $\mathbf{R}_{\mathbf{x}}$ can be exploited for multiple beams. In this case, for our 320 element array example, when around 200 beams are evaluated, the complexity of our CG-RDRCB will be similar to that of a full-dimension RCB under spherical uncertainty. However, our approach is still advantageous as, once $\hat{\mathbf{R}}_{\mathbf{x}}$ has been formed, the remaining operations needed for any of the proposed Krylov-RDRCBs can be performed separately for each beam, e.g., on separate processing units, yielding a highly parallelizable implementation, whereas the M-dimensional EVD needed by the full-dimension RCB cannot be easily parallelized. For RCBs under general non-degenerate uncertainty or when using spherical sets with extra linear constraints as in [19], a separate EVD has to be performed for every beam and therefore, the complexity savings of our methods are even greater. Compared to the CG-MVDR,

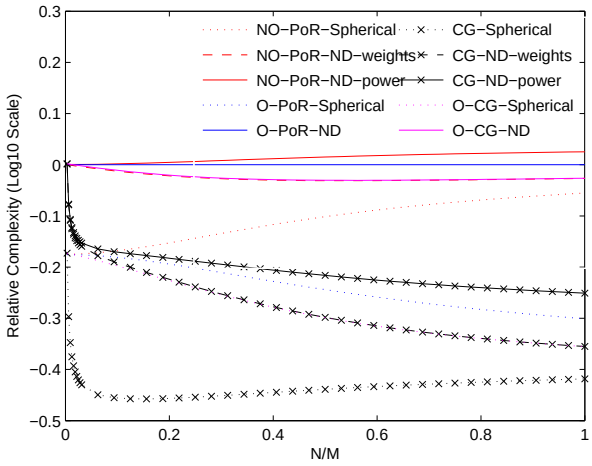


Fig. 1. Relative Krylov-RDRCB complexities. The NO-PoR-ND-weights and O-CG-ND curves are heavily overlapped, as are the O-CG-Spherical and CG-ND-weights curves.

which requires $\mathcal{O}(NM[M+6])$ operations[3], the CG RDRCBs exploiting spherical or non-degenerate

[3]In the CG-MVDR, $\mathcal{O}(NM)$ operations are needed to compute the weights in addition to the $\mathcal{O}(NM[M+5])$ operations needed for subspace expansion.

TABLE IV

ONLINE OPERATION COUNTS FOR NON-DEGENERATE POR-BASED RDRCBS.

	NO-PoR	O-PoR
Evaluate DRT	$\mathcal{O}(NM[M+1])$	$\mathcal{O}(NM[M+1+N])$
$\mathbf{R_y} = \mathbf{D}^H\mathbf{R_x}\mathbf{D}$	$\mathcal{O}(NM[M+N])$	
$\bar{\mathbf{b}} = \mathbf{D}^H\bar{\mathbf{a}}$	$\mathcal{O}(NM)$	
$(\mathbf{D}^H\mathbf{E}^{-1}\mathbf{D})$	$\mathcal{O}(NM[M+N])$	
$\mathbf{V\Gamma V}^H = \mathbf{D}^H\mathbf{E}^{-1}\mathbf{D}$	$\mathcal{O}(4\frac{1}{3}N^3)$	
$\mathbf{F}^{\frac{1}{2}} = \mathbf{\Gamma}^{-\frac{1}{2}}\mathbf{V}^H$	$\mathcal{O}(N^2)$	
$\breve{\mathbf{b}} = \mathbf{F}^{\frac{1}{2}}\bar{\mathbf{b}}$	$\mathcal{O}(N^2)$	
$\breve{\mathbf{R}}_\mathbf{y} = \mathbf{F}^{\frac{1}{2}}\mathbf{R_y}\mathbf{F}^{\frac{H}{2}}$	$\mathcal{O}(2N^3)$	
EVD($\breve{\mathbf{R}}_\mathbf{y}$)	$\mathcal{O}(4\frac{1}{3}N^3)$	
Solve for $\hat{\breve{\mathbf{b}}}$	$\mathcal{O}(2N[N+1]+n_{\text{iter}}6N)$	
$\hat{\mathbf{b}}_0 = \mathbf{F}^{-\frac{1}{2}}\hat{\breve{\mathbf{b}}} = \mathbf{V\Gamma}^{\frac{1}{2}}\hat{\breve{\mathbf{b}}}$	$\mathcal{O}(N[N+1])$	
$\mathbf{w}_{\text{RDRCB}}$	$\mathcal{O}(N[N+1])$	
Power $[\mathbf{D}^H\mathbf{D}]^{-1}$	$\mathcal{O}(N^2[M+N])$	0

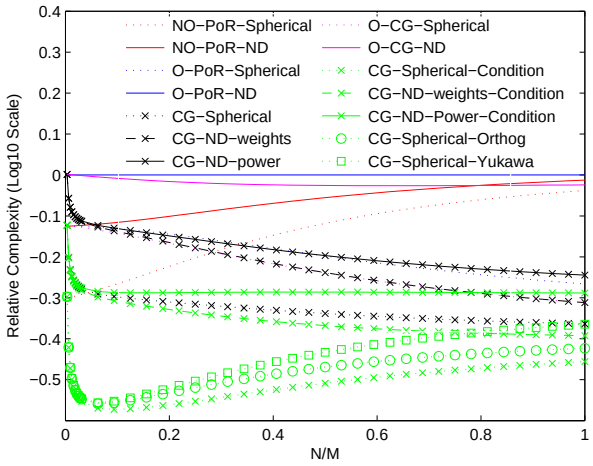


Fig. 2. Relative RDRCB complexities when exploiting the rank-selection rule in (28), unless appended with "Yukawa", "Condition" or "Orthog", which use rules (31), (32) and (33), respectively. The curves for O-PoR-Spherical and CG-ND-Power are heavily overlapped, as are the curves for O-CG-Spherical and CG-ND-weights.

uncertainty sets are only 1.2 or 2.2 times more complex, respectively, which is a small price to pay for the additional benefits arising from robustness to ASV errors.

TABLE V

ONLINE OPERATION COUNTS FOR CG-BASED RDRCBS.

	Spherical	Non-Degenerate
Evaluate DRT	$\mathcal{O}(NM[M+5])$	
$\mathbf{R}_{\mathbf{y}} = \mathbf{\Lambda}_{\text{CG}}$	$\mathcal{O}(M[M+1])$	
$\bar{\mathbf{b}} = \mathbf{D}^H \bar{\mathbf{a}}$	$\mathcal{O}(NM)$	
$\breve{\mathbf{b}} = \mathbf{\Lambda}_{\text{CG}}^{-\frac{1}{2}} \bar{\mathbf{b}}$	$\mathcal{O}(N)$	
Form $\mathbf{M}$	$\mathcal{O}(N^2[M+2])$	$\mathcal{O}(N[M^2+NM+2N])$
EVD($\mathbf{M}$)	$\mathcal{O}(4\frac{1}{3}N^3)$	
Solve for $\hat{\breve{\mathbf{b}}}$	$\mathcal{O}(2N[N+1]+\tilde{n}_{\text{iter}}6N)$	
$\hat{\mathbf{b}}_0 = \mathbf{\Lambda}_{\text{CG}}^{\frac{1}{2}} \hat{\breve{\mathbf{b}}}$	$\mathcal{O}(N)$	
$\mathbf{w}_{\text{RDRCB}}$	$\mathcal{O}(2N)$	
Power	$\mathcal{O}(3N)$	$\mathcal{O}(N^2[M+N])$ for $[\mathbf{D}^H\mathbf{D}]^{-1}$

VI. Rank Selection

Selecting the rank N is important for computational complexity and performance. Here, we examine the efficient implementation of two existing stopping criteria for selecting N and also propose two new rules for the CG-RDRCB. In [52], it was noted that the basis vectors in the NO-PoR transform (8) can become nearly linearly dependent (at least numerically) for quite small values of N. It was also noted that if a new potential basis vector belongs to the subspace spanned by the current transform, $\mathbf{D}$, i.e., if $\mathbf{g}_N = \hat{\mathbf{R}}_{\mathbf{x}}^N \bar{\mathbf{a}} \in \mathcal{R}(\mathbf{D})$, then $\mathbf{g}_n \in \mathcal{R}(\mathbf{D})$ for all $n \geq N$. Thus, the following stopping rule was suggested [52]

$$N = \max \left\{ n : \frac{\left\| \mathbf{\Pi}_{\mathbf{D}_{n-1}}^{\perp} \hat{\mathbf{R}}_{\mathbf{x}}^n \bar{\mathbf{a}} \right\|_2}{\left\| \hat{\mathbf{R}}_{\mathbf{x}}^n \bar{\mathbf{a}} \right\|_2} > \delta \right\}, \tag{28}$$

where the ratio represents, for a new potential Krylov basis vector, the fraction of its energy not contained in the current Krylov subspace and where δ is a small positive constant which imposes a minimum bound on this energy. We now consider how to implement the rule efficiently with each of the Krylov-subspace expansion techniques described in Section II-C. Calculating $\mathbf{g}_n = \hat{\mathbf{R}}_{\mathbf{x}}^n \bar{\mathbf{a}}$ requires $\mathcal{O}(M^2)$ operations and $\|\mathbf{g}_n\|_2$ requires $\mathcal{O}(M)$ operations. Noting that the square of the numerator in (28) can be written as $\|\mathbf{g}_n\|_2^2 - \mathbf{g}_n^H \mathbf{\Pi}_{\mathbf{D}_{n-1}} \mathbf{g}_n$, we need to calculate $\mathbf{g}_n^H \mathbf{\Pi}_{\mathbf{D}_{n-1}} \mathbf{g}_n$. Defining $\tilde{\mathbf{g}}_n = \mathbf{D}_{n-1}^H \mathbf{g}_n$, which costs $\mathcal{O}([n-1]M)$ operations to calculate, we can write $\mathbf{g}_n^H \mathbf{\Pi}_{\mathbf{D}_{n-1}} \mathbf{g}_n = \tilde{\mathbf{g}}_n^H \left[\mathbf{D}_{n-1}^H \mathbf{D}_{n-1}\right]^{-1} \tilde{\mathbf{g}}_n$. Since $\mathbf{D}_n \triangleq \begin{bmatrix} \mathbf{D}_{n-1} & \mathbf{d}_n \end{bmatrix}$, we can write $\left[\mathbf{D}_n^H \mathbf{D}_n\right]^{-1}$ efficiently in terms of $\left[\mathbf{D}_{n-1}^H \mathbf{D}_{n-1}\right]^{-1}$ by using the block matrix inversion lemma (see, e.g., [78]). Letting $\tilde{\mathbf{d}}_n = \mathbf{D}_{n-1}^H \mathbf{d}_n$, which requires $\mathcal{O}([n-1]M)$ operations and $\breve{\tilde{\mathbf{d}}}_n = \left[\mathbf{D}_{n-1}^H \mathbf{D}_{n-1}\right]^{-1} \tilde{\mathbf{d}}_n$, which requires $\mathcal{O}([n-1]^2)$ operations, we can write

$$\left[\mathbf{D}_n^H \mathbf{D}_n\right]^{-1} = \begin{bmatrix} \left[\mathbf{D}_{n-1}^H \mathbf{D}_{n-1}\right]^{-1} + \frac{\breve{\tilde{\mathbf{d}}}_n \breve{\tilde{\mathbf{d}}}_n^H}{\nu} & -\frac{\breve{\tilde{\mathbf{d}}}_n}{\nu} \\ -\frac{\breve{\tilde{\mathbf{d}}}_n^H}{\nu} & \frac{1}{\nu} \end{bmatrix}, \tag{29}$$

where $\nu = \|\mathbf{d}_n\|_2^2 - \tilde{\mathbf{d}}_n^H \breve{\tilde{\mathbf{d}}}_n$, which requires $\mathcal{O}(M+n-1)$ operations. Forming the upper left block of the matrix on the r.h.s of (29) requires an additional $\mathcal{O}([n-1]^2)$ operations. Thus, in general, updating $\left[\mathbf{D}_n^H \mathbf{D}_n\right]^{-1}$ requires $\mathcal{O}(nM + 2n^2 - 4n)$ operations, so that updating $\left[\mathbf{D}_{n-1}^H \mathbf{D}_{n-1}\right]^{-1}$ requires $\mathcal{O}([n-1]M + 2n^2 - 8n)$ operations at the nth iteration. For the NO-PoR method, $\mathbf{g}_n = \hat{\mathbf{R}}_{\mathbf{x}}^n \bar{\mathbf{a}}$ and $\|\mathbf{g}_n\|_2^2$ are available at each iteration and, as $[\mathbf{D}_N^H \mathbf{D}_N]^{-1}$ is computed in the implementation of the stopping rule, no additional operations are needed for power estimation.

For the O-PoR and O-CG methods, the columns of $\mathbf{D}_n$ are orthonormal so that the inverse $\left[\mathbf{D}_{n-1}^H \mathbf{D}_{n-1}\right]^{-1}$ does not need calculating. For the CG method, we note that the residuals $\{\mathbf{r}_i\}_{i=1}^{n-1}$, which are orthogonal to each other and are available at each iteration, span the same subspace as $\{\mathbf{d}_i\}_{i=1}^{n-1}$ and so, we can

write

$$\mathbf{g}_n^H \mathbf{\Pi}_{\mathbf{D}_{n-1}} \mathbf{g}_n = \bar{\mathbf{g}}_n^H \bar{\mathbf{\Lambda}}_{n-1}^{-1} \bar{\mathbf{g}}_n, \tag{30}$$

where $\bar{\mathbf{g}}_n = \bar{\mathbf{R}}_{n-1}^H \mathbf{g}_n$, with $\bar{\mathbf{R}}_{n-1} = \left[\begin{array}{ccc} \mathbf{r}_1 & \dots & \mathbf{r}_{n-1} \end{array} \right]$ and $\bar{\mathbf{\Lambda}}_{n-1} = \bar{\mathbf{R}}_{n-1}^H \bar{\mathbf{R}}_{n-1} = \text{diag}\left\{ \left[\begin{array}{ccc} \|\mathbf{r}_1\|_2^2 & \dots & \|\mathbf{r}_{n-1}\|_2^2 \end{array} \right] \right\}$ which is a diagonal matrix. For each of the methods, Table VI summarizes the additional operations required to calculate the function in (28) at the nth iteration, illustrating that for small N, NO-PoR requires the fewest additional operations. We remark that for both the CG and NO-PoR based RDRCBs under non-degenerate ellipsoidal uncertainty, extra operations are required to compute $\left[\mathbf{D}^H\mathbf{D}\right]^{-1}$ in the power estimation step. If (28) is implemented using (29), then $\left[\mathbf{D}^H\mathbf{D}\right]^{-1}$ will have already been computed. Thus, for the CG-RDRCB, if only the weights are required, then we implement (28) via (30); however, if power estimation is also required, then we use (29). Fig. 2 illustrates the relative complexities when exploiting the stopping rule in (28), indicating that the CG-based algorithms are still the least complex, except for very low values of N/M.

In [71], the following criterion was proposed

$$N = \max \left\{ n : \frac{\left\| \mathbf{\Pi}^{\perp}_{\hat{\mathbf{R}}_{\mathbf{x}}\mathbf{D}_{n-1}} \bar{\mathbf{a}} \right\|_2}{\|\bar{\mathbf{a}}\|_2} > \delta \right\}, \tag{31}$$

where it is was found to possess more natural $N - \delta$ relations than (28), which enable a simpler selection of δ. We refer the reader to [71] for more details on this aspect and where it is also shown that (28) and (31) can be derived from two necessary and sufficient conditions for the low-rank subspace to contain the optimal (MVDR-type) filter. Here, we need to calculate $\bar{\mathbf{a}}^H \mathbf{\Pi}_{\hat{\mathbf{R}}_{\mathbf{x}}\mathbf{D}_{n-1}} \bar{\mathbf{a}}$. At each iteration, $\breve{\mathbf{D}}_{n-1} \triangleq \hat{\mathbf{R}}_{\mathbf{x}}\mathbf{D}_{n-1}$ needs updating. We will already have all but the last column of this matrix from previous iterations, thus we need only calculate $\hat{\mathbf{R}}_{\mathbf{x}}\mathbf{d}_{n-1}$, which can be calculated at no extra cost in all of the algorithms. Note that $\bar{\mathbf{a}}^H \mathbf{\Pi}_{\hat{\mathbf{R}}_{\mathbf{x}}\mathbf{D}_{n-1}} \bar{\mathbf{a}} = \bar{\mathbf{a}}^H \breve{\mathbf{D}}_{n-1} \left[\breve{\mathbf{D}}_{n-1}^H \breve{\mathbf{D}}_{n-1} \right]^{-1} \breve{\mathbf{D}}_{n-1}^H \bar{\mathbf{a}}$. In calculating $\breve{\mathbf{D}}_{n-1}^H \bar{\mathbf{a}}$, we only need to calculate $\breve{\mathbf{d}}_{n-1}^H \bar{\mathbf{a}}$, where $\breve{\mathbf{d}}_{n-1}$ denotes the $(n-1)$th column of $\breve{\mathbf{D}}_{n-1}$, as the other parts have been calculated at previous iterations. We can use the same approach used above to update $\left[\mathbf{D}_{n-1}^H \mathbf{D}_{n-1}\right]^{-1}$ efficiently to update $\left[\breve{\mathbf{D}}_{n-1}^H \breve{\mathbf{D}}_{n-1} \right]^{-1}$. Therefore, at the nth iteration, the cost of implementing the rule in (31) is $\mathcal{O}(nM + 3n^2 - 9n)$. Thus, for small N, it is cheaper to use (31) rather than (28). When exploiting (31), there is no difference from the relative complexities plotted in Fig. 1.

The operation counts in Table VI shows that after N iterations, more than $\mathcal{O}(NM[M+2])$ additional operations are needed to implement (28) with the CG (or O-PoR) subspace expansion method, which is a significant proportion of the overall complexity. Here, we propose two alternative rules, which can be implemented more efficiently with the CG-RDRCB. Firstly, we note that using (28) prevents

$\mathbf{R}_\mathbf{y} = \mathbf{D}^H \mathbf{R}_\mathbf{x} \mathbf{D}$ from becoming poorly conditioned. Therefore, one approach is to threshold the condition number of $\mathbf{R}_\mathbf{y}$. When exploiting the CG direction vectors to form $\mathbf{D}$, the resulting $\mathbf{R}_\mathbf{y} = \mathbf{\Lambda}_{\text{CG}}$ is diagonal so that its eigenvalues are exactly the non-zero entries in $\mathbf{\Lambda}_{\text{CG}}$. Thus, we propose the following stopping rule

$$N = \max \left\{ n : \frac{\max\{\boldsymbol{\lambda}_{n,\text{CG}}\}}{\min\{\boldsymbol{\lambda}_{n,\text{CG}}\}} < \delta_{\text{CG}} \right\}, \tag{32}$$

where $\boldsymbol{\lambda}_{n,\text{CG}}$ contains the n diagonal elements of $\mathbf{\Lambda}_{\text{CG}}$ (18) and δ_{CG} is an upper bound on the allowed condition number of $\mathbf{R}_\mathbf{y}$. Since we always calculate $\mathbf{\Lambda}_{\text{CG}}$ in the CG-RDRCB, there is negligible additional complexity in exploiting this rule.

We also propose an alternative rule that ensures that the $\hat{\mathbf{R}}_\mathbf{x}$-orthogonality of the direction vectors is maintained. In theory, each new CG direction vector should be $\hat{\mathbf{R}}_\mathbf{x}$-orthogonal to all previous direction vectors, i.e., $\mathbf{d}_N^H \hat{\mathbf{R}}_\mathbf{x} \mathbf{d}_i = 0$ for $i = 1, \ldots, N-1$, however, empirically, we have found that even for quite small values of N this orthogonality can break down. Thus, we propose the following stopping rule

$$N = \max \left\{ n : \frac{1}{n-1} \sum_{i=1}^{n-1} \mathbf{d}_n^H \hat{\mathbf{R}}_\mathbf{x} \mathbf{d}_i < \delta_{\text{orthog}} \right\}. \tag{33}$$

Since $\left\{ \hat{\mathbf{R}}_\mathbf{x} \mathbf{d}_i \right\}_{i=1}^{n-1}$ are already available [see (10)], only $\mathcal{O}([n-1]M)$ additional operations are needed to implement the rule at the nth iteration, which is less than the rules in (28) and (31). Fig. 2 shows complexity plots when using all of the different rules with the CG-RDRCBs, at least under spherical uncertainty, indicating that (32) and (33) are the least complex. We remark that the curves exploiting the condition test rule (32) give an indication of the complexity when not using additional rank selection.

TABLE VI

ONLINE OPERATION COUNTS FOR COMPUTING RANK-SELECTION USING (28) AT THE nTH ITERATION.

	NO-PoR	O-PoR / O-CG	CG
$\mathbf{g}_n$	0	$\mathcal{O}(M^2)$	
$\|\mathbf{g}_n\|_2^2$	0	$\mathcal{O}(M)$	
$\tilde{\mathbf{g}}_n$ or $\bar{\mathbf{g}}_n$	$\mathcal{O}([n-1]M)$		
$\left[\mathbf{D}_{n-1}^H \mathbf{D}_{n-1}\right]^{-1}$ or $\bar{\mathbf{\Lambda}}_{n-1}^{-1}$	$\mathcal{O}([n-1]M + 2n^2 - 8n)$	0	$\mathcal{O}(M)$
$\tilde{\mathbf{g}}_n^H \left[\mathbf{D}_{n-1}^H \mathbf{D}_{n-1}\right]^{-1} \tilde{\mathbf{g}}_n$ or $\bar{\mathbf{g}}_n^H \bar{\mathbf{\Lambda}}_{n-1}^{-1} \bar{\mathbf{g}}_n$	$\mathcal{O}(n^2 - n)$	$\mathcal{O}(n)$	

VII. Numerical Examples

Here, we examine numerical examples for a large $M = 320$ element half wavelength spaced planar array, with $M_h = 40$ elements in each row and $M_v = 8$ rows. We focus on the performance in snapshot deficient scenarios, where $K << M$ snapshots are available. The simulated scenario is relevant in, e.g., passive sonar, where large multidimensional arrays are often employed to detect and localize sources (weak and strong) in an environment where there are noisy merchant ships and/or jammers, isotropic ambient noise and sensor noise (see, e.g., [2], [21], [29]). The data were simulated using the array covariance matrix $\mathbf{R_x} = \sigma_0^2 \mathbf{a}_0 \mathbf{a}_0^H + \mathbf{Q_x}$, where σ_0^2 and $\mathbf{a}_0$ denote the SOI power and the SOI ASV, respectively, and where $\mathbf{Q_x} = \sum_{i=1}^{d} \sigma_i^2 \mathbf{a}_i \mathbf{a}_i^H + \sigma_s^2 \mathbf{I} + \sigma_{\text{iso}}^2 \mathbf{Q}_{\text{iso}}$. Finite-data snapshots were produced by coloring independent zero-mean complex circularly symmetric normal random vectors, with covariance equal to the identity matrix, using $\mathbf{R}_{\mathbf{x}}^{\frac{1}{2}}$. The noise plus interference covariance $\mathbf{Q_x}$ consists of terms due to d zero-mean uncorrelated interfering sources, where for the ith interferer σ_i^2 and $\mathbf{a}_i$ denote the source power and ASV, a term modeling sensor noise $\sigma_s^2 \mathbf{I}$, with sensor noise power σ_s^2, and a term modeling an isotropic ambient noise $\sigma_{\text{iso}}^2 \mathbf{Q}_{\text{iso}}$, with power σ_{iso}^2. The isotropic noise covariance is given by $[\mathbf{Q}_{\text{iso}}]_{m,n} = \text{sinc}[\pi g_{mn}]$, where g_{mn} is the distance between the mth and nth sensors in units of wavelength. The ith source (SOI or interference) ASV is simulated according to $\mathbf{a}_i = \mathbf{a}(\bar{\boldsymbol{\theta}}_i + \boldsymbol{\delta}_i) + \sigma_{e,i} \mathbf{e}_i$, where $\bar{\boldsymbol{\theta}}_i = \left[\bar{\theta}_i, \bar{\phi}_i\right]^T$ denotes the assumed AOA, comprising the assumed azimuth angle $\bar{\theta}_i$ and the assumed elevation angle $\bar{\phi}_i$, $\boldsymbol{\delta}_i = [\delta_{\theta_i}, \delta_{\phi_i}]^T$ denotes the AOA mismatch, comprising the mismatch in azimuth angle δ_{θ_i} and the mismatch in elevation angle δ_{ϕ_i}, $\sigma_{e,i}$ denotes the length of the arbitrary error vector and $\mathbf{e}_i$ is a zero-mean complex circularly symmetric random vector with unit norm. When $\boldsymbol{\delta}_i \neq 0$ an AOA error exists and when $\sigma_{e,i} \neq 0$, an arbitrary ASV error exists. In our application of interest, we are interested in forming multiple beams to survey an angular region and assume that the beams are cosine-spaced at $1/M_h$ and $1/M_v$ intervals in azimuth and elevation, respectively. We assume that the SOI beam corresponds to the beam whose center is defined by $\bar{\theta}_0 = 89.86°$ and $\bar{\phi}_0 = 94.78°$. When simulating the AOA of the SOI, we assume that its azimuth angle lies anywhere in the interval $[\theta_l, \theta_u]$ with uniform probability, where $\theta_l(\theta_u)$ is the angle midway between the center of SOI azimuth beam and the center of the adjacent beam with lower (higher) azimuth angle, and that its elevation angle lies anywhere in the interval $[\phi_l, \phi_u]$ with uniform probability, where $\phi_l(\phi_u)$ is the angle midway between the center of SOI elevation beam and the center of the adjacent beam with lower (higher) elevation angle; $\boldsymbol{\delta}_0$ is set accordingly. In the following, unless otherwise stated, $\sigma_0^2 = 10$ dB, $\sigma_{e,0} = 1$, $\sigma_s^2 = 0$ dB, $\sigma_{\text{iso}}^2 = 1$ dB and $d = 10$. Except for $\sigma_5^2 = 30$ dB and $\sigma_6^2 = 30$ dB, the interference powers are set to

20 dB. The azimuth angles of the interferers are set to $\bar{\theta}_1 = 10°$, $\bar{\theta}_2 = 20°$, $\bar{\theta}_3 = 70°$, $\bar{\theta}_4 = 85°$, $\bar{\theta}_5 = 95°$, $\bar{\theta}_6 = 100°$, $\bar{\theta}_7 = 115°$, $\bar{\theta}_8 = 130°$, $\bar{\theta}_9 = 150°$ and $\bar{\theta}_{10} = 160°$. Except for $\bar{\phi}_1 = 130°$, $\bar{\phi}_2 = 112°$ and $\bar{\phi}_3 = 75°$, the elevation angles of the interferences are set to $94.78°$. Furthermore, $\{\boldsymbol{\delta}_i = \mathbf{0}\}_{i=1}^{10}$ and $\{\sigma_{e,i} = 1\}_{i=1}^{10}$.

We now discuss the full-dimension (or element-space) uncertainty sets that we have used in the (RD)RCBs. RCBs are most commonly used with spherical sets, of which the smallest possible spherical uncertainty set [13], [14] is a particularly tight set that offers good control of AOA uncertainty without sacrificing as much in interference rejection as other spherical set designs [74]. In [74], non-degenerate ellipsoids, each formed from a flat minimum volume ellipsoid (MVE) and a *spherical* error ellipsoid, were proposed for azimuth and elevation beamforming of multidimensional arrays. These sets, termed non-degenerate MVE (NDMVE) sets, allow for better control of azimuth and elevation uncertainty and can give better interference rejection than the associated (tight) spherical sets. Here, we exploit (tight) spherical sets and NDMVE sets, not to endorse one over the other, but to illustrate that our algorithms work with both spherical and more general non-degenerate sets. In the interest of brevity, we refer the reader to [74] for details on how to form these sets, which are designed for each azimuth and elevation pair, assuming the beams are spaced as described above. We remark that the (tight) spherical uncertainty set radius (squared) for the SOI beam was calculated as $\epsilon = 120$. In designing the NDMVE sets, we set $1 - \gamma = -80$ dB in (28) in [74] and used a *spherical* error ellipsoid radius of $\breve{\epsilon} = 10$ in (34) in [74].

In the following, we examine the delay-and-sum (DAS), the MVDR, the RCB-NDMVE and the RCB-Spherical full-dimension beamformers; the following RDRCBs: NO-PoR-Spherical, NO-PoR-NDMVE, O-PoR-Spherical, O-PoR-NDMVE, O-CG-Spherical, O-CG-NDMVE, CG-Spherical and CG-NDMVE, where we append Spherical or NDMVE to denote that we are exploiting a tight spherical uncertainty set or a non-degenerate NDMVE uncertainty set; and the following standard, non-robust RDMVDRs: NO-PoR-MVDR, O-PoR-MVDR, O-CG-MVDR and CG-MVDR. The RDMVDR weights can be formed by using (15) with $\hat{\mathbf{b}}_0$ replaced with $\bar{\mathbf{b}}$. Unless otherwise stated, we use $K = 80$ snapshots for weight estimation. When $K < M$, the inverse of $\hat{\mathbf{R}}_{\mathbf{x}}$ in (2) does not exist and therefore, we use a low-rank pseudo-inverse, which calculates the inverse based on the eigenpairs associated with the K non-zero eigenvalues. We evaluate performance via the signal-to-interference-plus-noise ratio (SINR) metric, defined as $\text{SINR} = \sigma_0^2 \frac{|\mathbf{w}^H \mathbf{a}_0|^2}{\mathbf{w}^H \mathbf{Q}_{\mathbf{x}} \mathbf{w}}$. It is well known that the optimal SINR is given by $\text{SINR}_{\text{opt}} = \sigma_0^2 \mathbf{a}_0^H \mathbf{Q}_{\mathbf{x}}^{-1} \mathbf{a}_0$. In our Monte-Carlo simulations, $\mathbf{a}_0$ and $\mathbf{Q}_{\mathbf{x}}$ vary due to modeling the SOI and interferer ASV errors and therefore, we plot the optimal SINR as the mean of SINR_{opt} over the Monte-Carlo simulations.

A. SINR versus Rank

Here, we investigate how the performances of the reduced-dimension algorithms vary with the rank N. Fig. 3 (a) shows the SINR vs. N for a strong 60 dB SOI, illustrating that up to rank-3, the different Krylov-based variants of the RDMVDR, the spherical RDRCB and the non-degenerate RDRCB give the same performance. The NO-PoR and O-CG/CG algorithm SINRs diverge after ranks 3 and 7, respectively. This divergence is due to numerical instability and is related to the condition number of $\mathbf{R_y}$. Fig. 3 (c) shows the mean condition number of $\mathbf{R_y}$ vs. N for the strong 60 dB SOI case, illustrating that when this exceeds a certain threshold (around 160 dB), the associated algorithm performance breaks down. Providing this threshold is not breached, the Krylov-RDRCBs are able to maintain near optimum performance. However, the Krylov-RDMVDRs deteriorate after rank-1 due to severe SOI cancellation. Fig. 3 (b) shows the SINR vs. N for a weaker 20 dB SOI, for which the O-PoR, O-CG and CG variants give the same performances for all of the ranks tested. The NO-PoR variant starts diverging after rank-7, which, as shown by Fig. 3 (d) for the weaker 20 dB SOI, is caused by numerical instability, shown by the mean condition number of $\mathbf{R_y}$ exceeding the 160 dB threshold. Thus, providing that the condition number of $\mathbf{R_y}$ does not exceed a certain threshold, then for a fixed rank N, the performance of the non-degenerate Krylov-RDRCBs are equal to each other, the performance of the spherical Krylov-RDRCBs are equal to each other and the performance of the Krylov-RDMVDRs are equal to each other. The results in Fig. 3 (a)–(d) show how the CG, O-CG and O-PoR subspace expansion methods are significantly more numerically stable than the NO-PoR method. They also show that the Krylov-RDMVDRs are highly sensitive to the rank and particularly to its over-determination, whereas, providing that numerical stability is maintained, the Krylov-RDRCBs are relatively insensitive to rank over-determination, which represents a significant advantage for the RDRCBs.

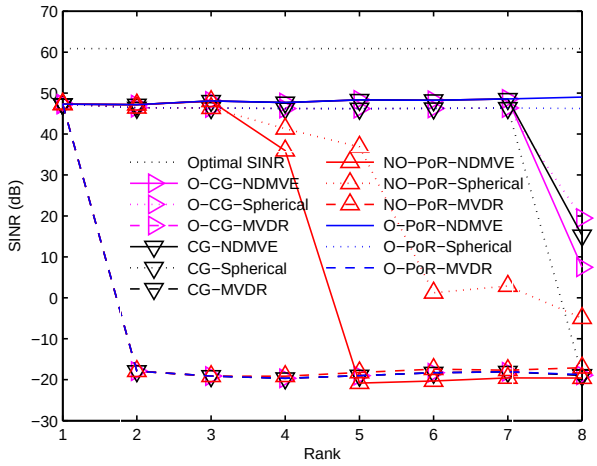
When using the stopping rules, we examine the SINR as a function of the maximum allowable rank. If the maximum allowable rank is set equal to N, the stopping rule may set the rank to any value less than or equal to N. As suggested in [52], we select $\delta = 0.01$ in Honig's rule (28). Based on the results in [71] and noting our level of snapshot support, we have used $\delta = 0.8$ with Yukawa's rule (31). Based on some empirical observations, we have selected $\delta_{\text{CG}} = 40$ dB for the condition number based rule (32) and $\delta_{\text{orthog}} = 1 \times 10^{-5}$ for the orthogonality testing based rule (33). Figs. 3 (e) and (f) show results with the CG-based algorithms for a 60 dB SOI and a 20 dB SOI, respectively. We note that when using Honig's rule (28) or Yukawa's rule (31) with these settings, the different Krylov-subspace variants give the same results, that is, all the Krylov-RDMVDRs give equal results, all the non-degenerate Krylov-RDRCBs

give the same results, and all the spherical Krylov-RDRCBs give the same results. The reason for this is that the stopping rules prevent any of the algorithms from becoming numerically unstable and as they all implement the same stopping rule, it means that each select the same rank and therefore, expand identical subspaces to give identical results. Therefore, we have only shown the results for the CG-based algorithms in Figs. 3 (e) and (f). Fig. 3 (e) shows that at high SNR, the stopping rules allow the CG-RDRCBs to maintain near-optimal performance by preventing degradation due to numerical instability. Since the CG-MVDR performance deteriorates for any rank greater that 1, only Yukawa's rule and the condition number based rule (32), which selected the rank as 1, enabled the CG-MVDR to perform well. Fig. 3 (f) shows that, for the 20 dB SOI, there was little difference between using a stopping rule or not with the CG-RDRCBs. Only the CG-NDMVE benefited slightly from using Honig's rule. Yukawa's rule allowed the CG-MVDR to maintain a reasonable performance, though, this was not as good as that obtained by the CG-RDRCBs when using another rule or a fixed rank.

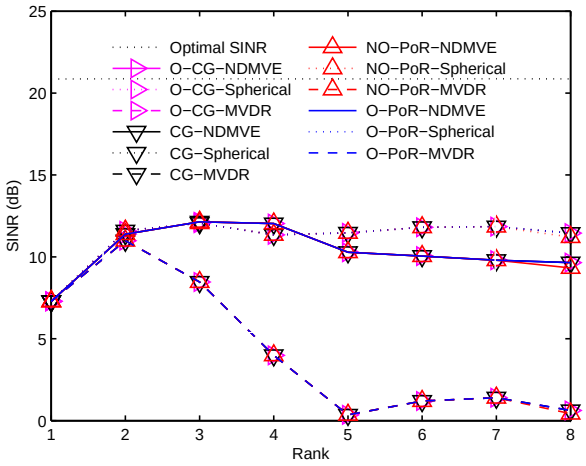
B. SINR versus SOI Power

Here, we examine the SINR as a function of the SOI power, which is shown in Fig. 4 (a) for fixed rank $N = 5$. For low SOI powers, below 0 dB, the reduced-dimension algorithms perform the same as each other and much better than the full-dimension algorithms. At low SOI powers, the DAS performs poorly as it is unable to form data-adaptive nulls towards the interferences. MVDR and RCB-NDMVE perform better than the DAS, but still perform poorly at these low SNRs due to their poor robustness to snapshot deficiency. RCB-Spherical performs much better, as it is inherently more robust to snapshot deficiency than both MVDR and RCB-NMVDE, however, it still performs worse at these lower SNRs than the proposed algorithms, which converge a lot faster. As the SNR increases, we observe how the performances of all of the MVDR-based algorithms, whether reduced-rank or not, start to deteriorate, as they are not robust to ASV errors. The robust algorithms mitigate this loss in performance. At the highest SOI power, the NO-PoR based algorithms start to diverge due to numerical instability, as discussed earlier.

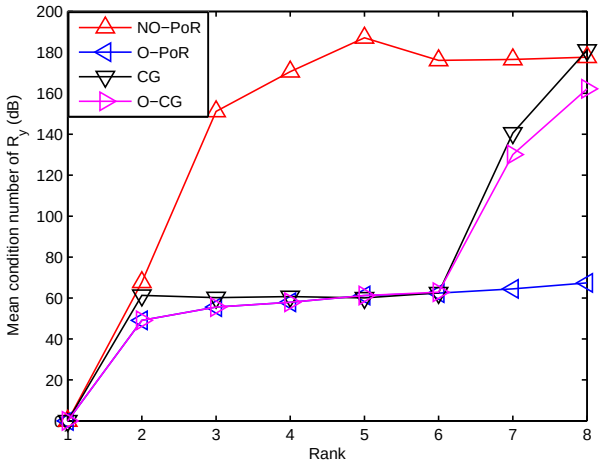
Fig. 4 (b) shows results when using the four stopping rules with the CG-based algorithms. With all the rules, except Yukawa's rule (31), the performances of the RDRCBs are similar to the fixed rank $N = 5$ case, though there is a slight loss in performance at low SOI power. With Yukawa's rule (31), there is a drop in SINR for the RDRCBs at 20 dB SOI power; however, the performances of the RDMVDRs now match the RDRCBs. With these settings, for the CG-MVDR, thresholding the condition number improves the performance, but there is little difference when using Honig's rule or the rule testing for orthogonality breakdown. We remark that increasing the value of Honig's threshold to improve the performance of



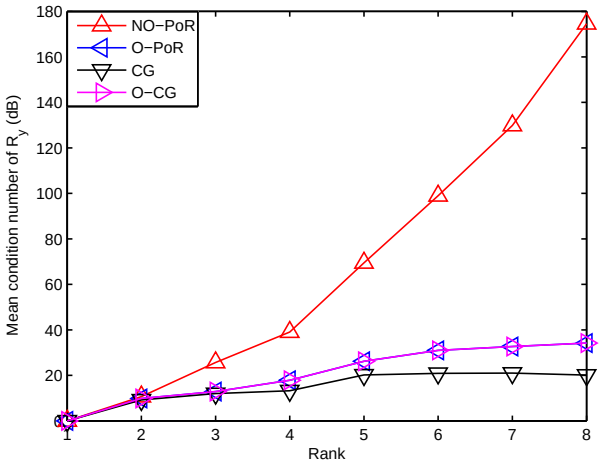
(a)



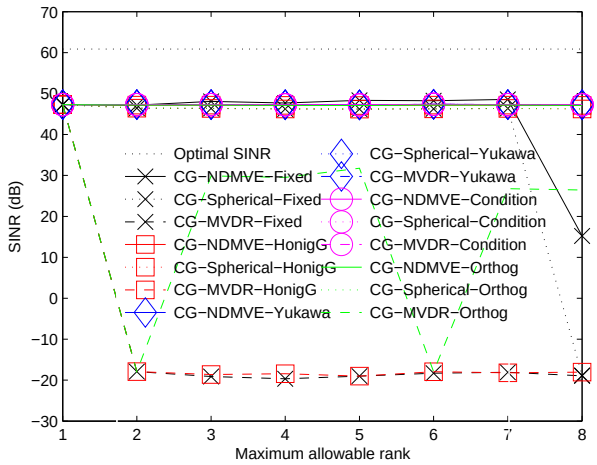
(b)



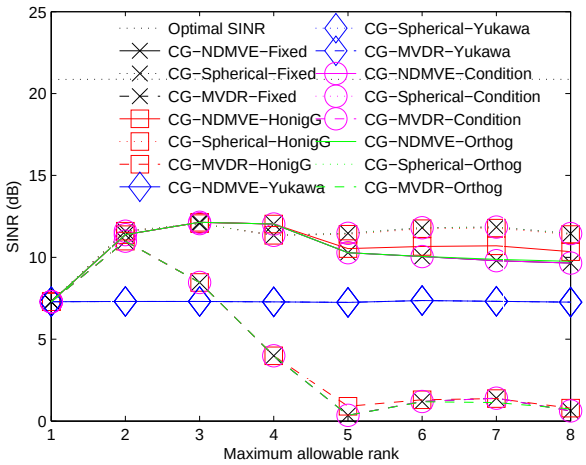
(c)



(d)



(e)



(f)

Fig. 3. For a SOI with AOA errors, $\sigma_{e,i} = 1$ and $K = 80$, SINR vs rank for (a) a 60 dB SOI, (b) a 20 dB SOI; mean condition number of $\mathbf{R}_y$ vs rank for (c) a 60 dB SOI or (d) a 20 dB SOI; SINR vs. maximum allowable rank, when using stopping rules, for (e) a 60 dB SOI or (f) a 20 dB SOI.

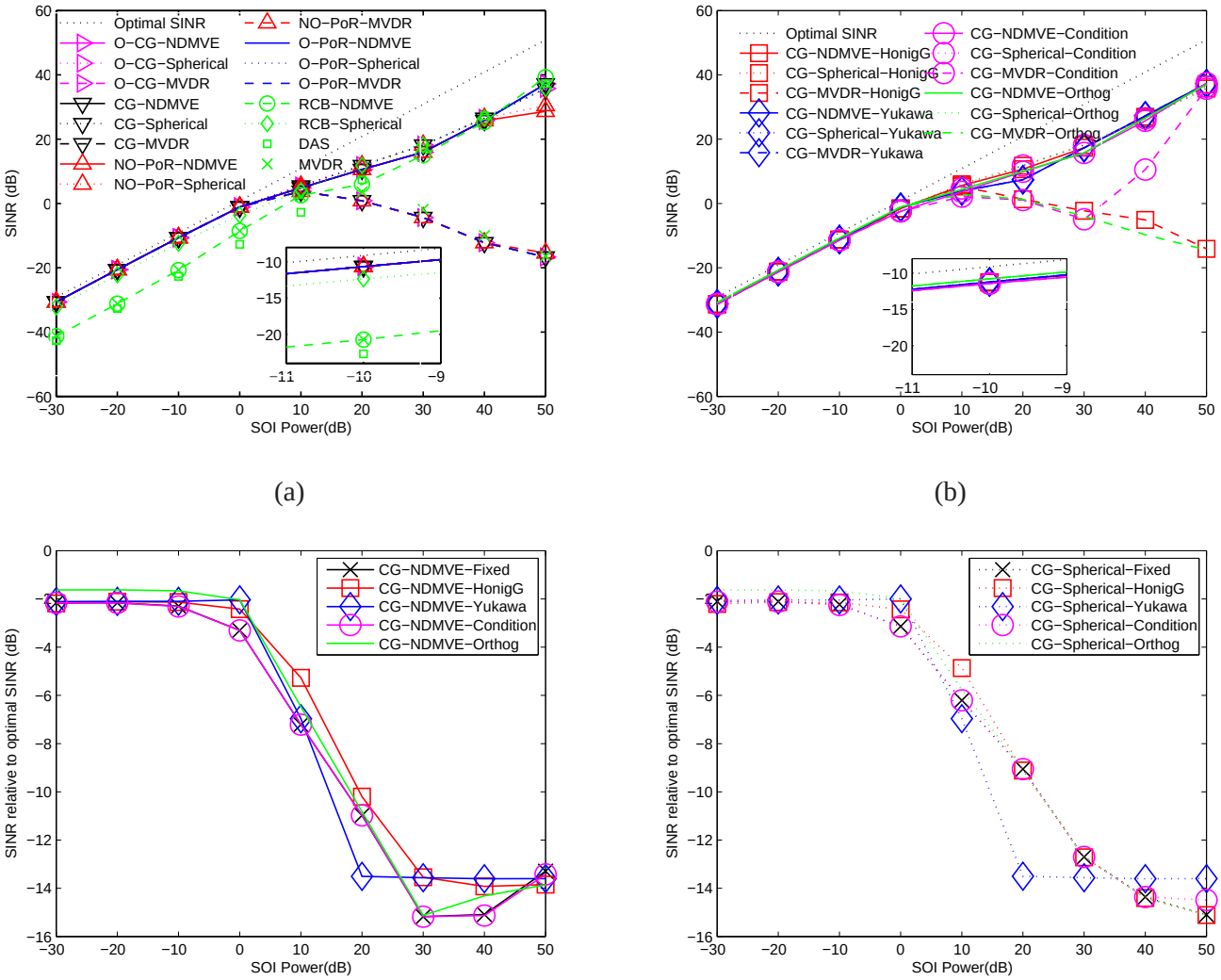


Fig. 4. SINR vs SOI power with $K = 80$ for (a) $N = 5$; for stopping rules with $N_{\text{Max}} = 8$ and $\delta = 0.01$ in (28), $\delta = 0.8$ in (31), $\delta_{\text{CG}} = 40$ dB in (32) and $\delta_{\text{orthog}} = 1 \times 10^{-5}$ in (33), (b) SINR vs SOI power, and SINR loss, relative to the optimal SINR, vs. SOI power for the CG-RDRCBs, where the fixed rank version uses $N = 8$, for (c) non-degenerate uncertainty, or (d) spherical uncertainty.

the RDMVDRs at high SNR reduces the performance at low SNR. Similarly, lowering the value of Yukawa's threshold can improve the performance of the RDRCBs at 20 dB, but then the performances of the RDMVDRs deteriorates. The performance of the CG-MVDR is highly sensitive to the selection of the rank and can benefit from rank-selection, especially Yukawa's rule, whereas the CG-RDRCBs are not. Figs. 4 (c) and (d) show the SINR loss, compared to optimal, vs the SOI power when using

different types of rank selection with the CG-RDRCBs under non-degenerate and spherical uncertainty, respectively. The performances of the efficient rules based on either thresholding the condition number or on testing the orthogonality of the CG direction vectors, are similar to using Honig and Goldstein's rule in (28), but can be implemented at a fraction of the complexity. The results also show that only minor additional performance gains over simply selecting the rank as the maximum rank of $N = 8$ can be achieved with stopping rules, as the RDRCBs are already robust to rank over-determination.

C. SINR versus the Number of Snapshots

Fig. 5 (a) illustrates the SINR versus the number of snapshots K for a strong 50 dB SOI, when we have selected $N = 5$ in the reduced-dimension algorithms. The CG, O-CG and O-PoR RDRCBs behave identically to each other and similar to the DAS, which we expect to behave well for this high SNR case, and much better than the standard MVDR algorithm and the associated non-robust RDMVDRs. The NO-PoR RDRCBs perform worse due to numerical instability. Fig. 5 (b) illustrates the results for a low SNR case with a SOI power of -30 dB, indicating that all of the methods exploiting data-dependent dimensionality reduction perform similarly, outperforming the full-dimension RCBs, DAS and MVDR. We remark that the SINR for RCB-Spherical does look good at these low levels of K, but it should be noted that its SINR is actually falling as K is increasing. This phenomenon has been seen elsewhere [2] and we remark that it is only for $K > M$, that SINR is expected to increase monotonically with K.

For the same data as used in Figs. 5 (a) and (b), Figs. 5 (c) and (d) show results for the CG-based algorithms exploiting different rank selection methods, showing that there is little difference in the behaviors of the CG-RDRCBs when using the different stopping rules, mainly as they are robust to model over-determination, whereas large improvements are possible in the CG-RDMVDR if, e.g., Yukawa's rule is used.

D. Spatial Spectra

In several applications, e.g., passive sonar, beamforming is used for imaging (in addition to increasing SINR) where good quality spatial spectra with good power estimates are desirable. Here, we examine the spatial spectra, assuming the azimuth and elevation beam spacings described earlier. Due to the similarity of the spatial spectra obtained using the CG, O-CG, O-PoR and NO-PoR methods, we only examine the CG-based algorithms here. Fig. 6 (a) shows the DAS spatial spectrum, where the source AOAs that appear in the scanning region are shown using red circles. Figs. 6 (b)–(f) show the azimuth spectra for the elevation beam containing the SOI (centered at 94.78°) when using different forms of rank selection.

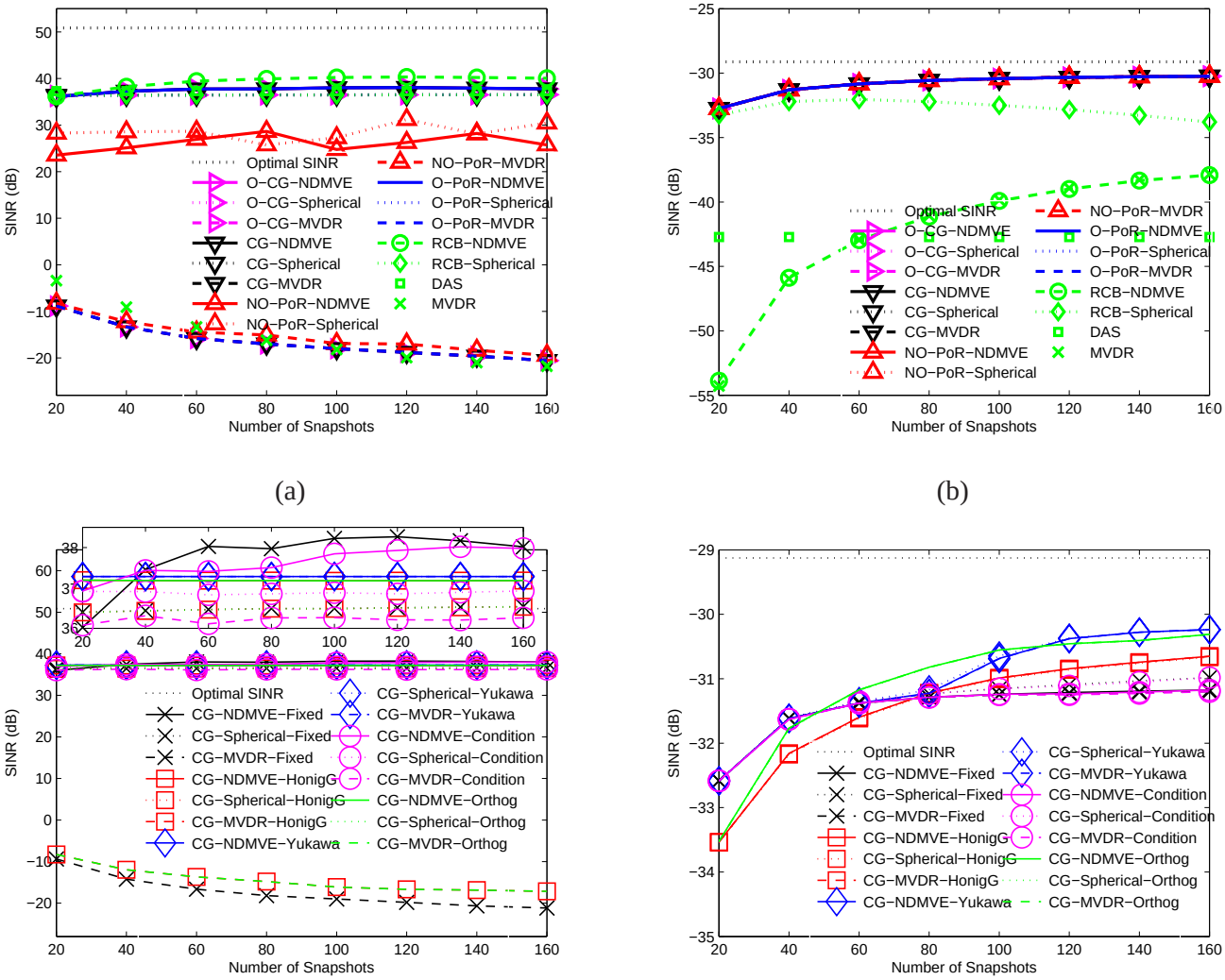
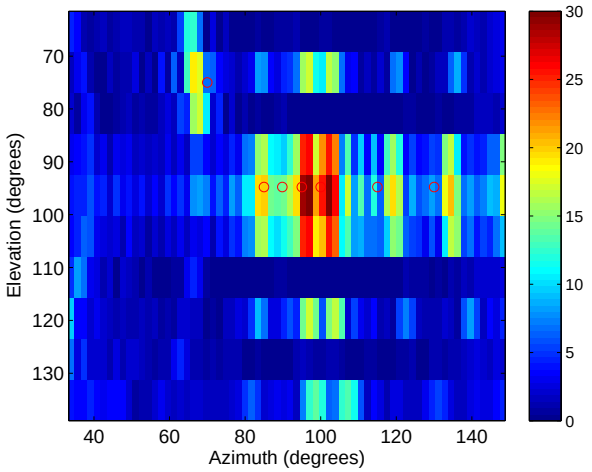
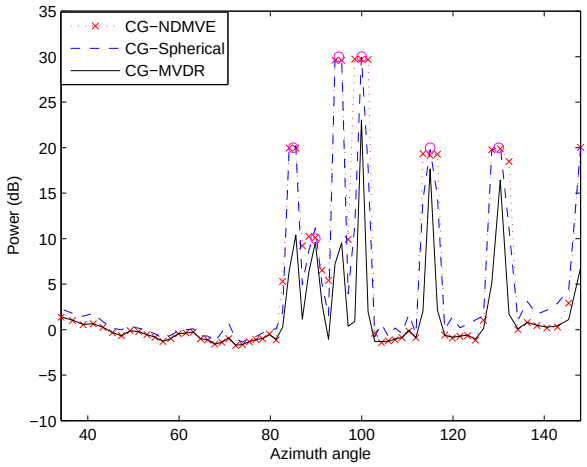


Fig. 5. SINR vs K, for a SOI with AOA errors and arbitrary ASV errors with $\sigma_e^2 = 1$, when $N = 5$ for (a) a 50 dB SOI and (b) a -30 dB SOI; when using $N = 8$ or stopping rules with $N_{\text{Max}} = 8$ and $\delta = 0.01$ in (28) or $\delta = 0.8$ in (31) or $\delta_{\text{CG}} = 40$ dB in (32) or $\delta_{\text{orthog}} = 1 \times 10^{-5}$ in (33) for (c) a 50 dB SOI and (d) a -30 dB SOI.

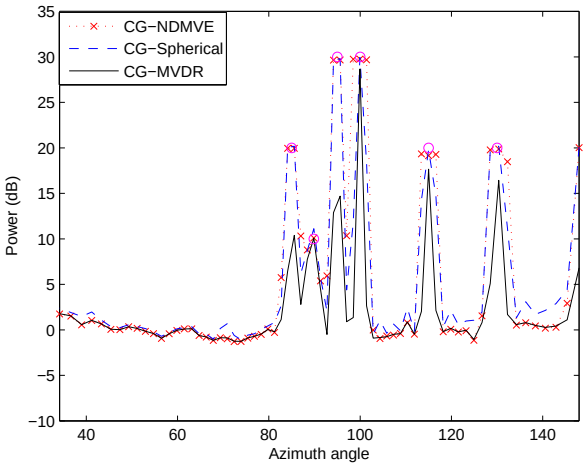
In all cases, the CG-RDRCBs provide robustness against SOI cancellation and give good quality power estimates, whereas the non-robust CG-MVDR suffers severe SOI cancellation in all cases except when using Yukawa's stopping rule (31).



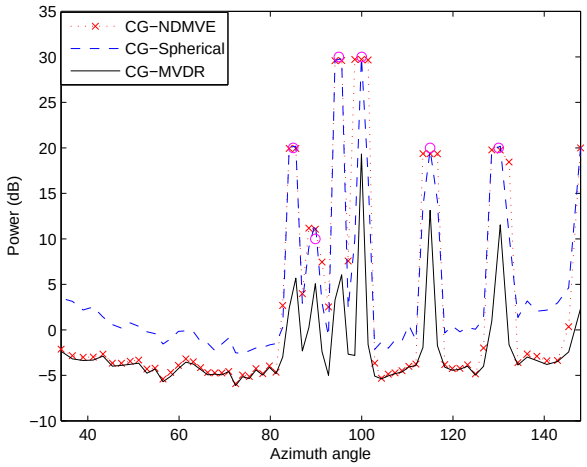
(a)



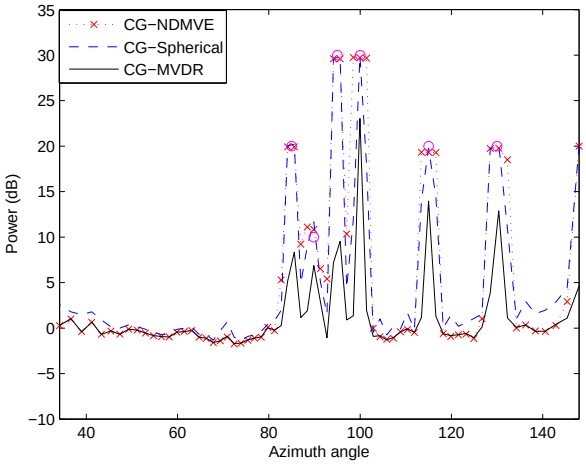
(b)



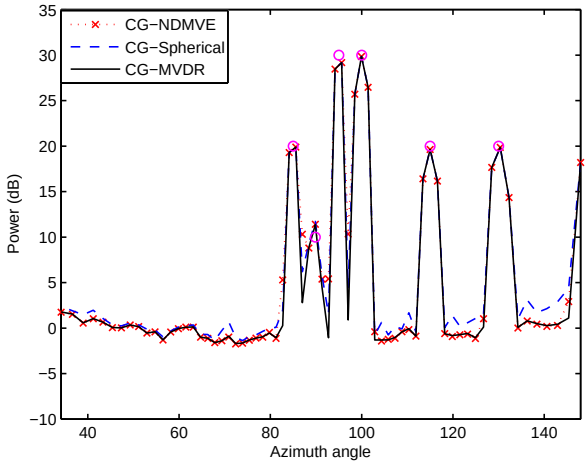
(c)



(d)



(e)



(f)

Fig. 6. Spatial spectra for (a) DAS; the CG-based algorithms with (b) $N = 5$, or when using stopping rules with $N_{\max} = 8$ and (c) (28) with $\delta = 0.01$, (d) (32) with $\delta_{\text{CG}} = 40$ dB, (e) (33) with $\delta_{\text{orthog}} = 10^{-5}$, or (f) (31) with $\delta = 0.8$.

E. Summary

From the results above we observe that

- A Krylov-RDRCB's performance is lower-bounded by its associated Krylov-RDMVDR performance.
- The O-PoR and CG/O-CG Krylov-subspace expansion methods are significantly more numerically stable than the NO-PoR method.
- Provided that numerical stability is maintained, the Krylov-RDRCBs give the same results.
- Providing that numerical stability is maintained, the Krylov-RDRCBs are insensitive to rank over-determination and SOI ASV errors, whereas the Krylov-RDMVDRs are not.
- Stopping rules can be used to maintain numerical stability and can provide further small performance enhancements in the Krylov-RDRCBs.
- A "good" stopping rule, such as Yukawa's rule, allows the performance of a Krylov-RDMVDR to be significantly improved.
- Krylov-RDRCBs converge faster than full-dimension RCBs and are significantly less complex.
- Krylov-RDRCBs provide good quality spatial spectra with good quality power estimates.

VIII. Conclusions

We have derived a family of low-complexity, quickly converging, Krylov-subspace reduced-dimension robust Capon beamformers (Krylov-RDRCBs) that combine data-dependent Krylov-subspace dimensionality reduction, computed using the Power-of-R (PoR) or conjugate gradient (CG) methods, with ellipsoidal uncertainty set based robust Capon beamformer (RCB) techniques. To derive the Krylov-RDRCBs, we have extended a recent RDRCB framework to allow for faster online computation of the reduced-dimension ellipsoids needed after rank reduction. The extended framework is applicable to all forms of dimensionality reduction and not just the Krylov methods considered in this paper. Existing Krylov-subspace techniques often suffer severe signal-of-interest cancellation, due to steering vector mismatch, and are extremely sensitive to rank selection, particularly to rank over-determination. The proposed Krylov-RDRCBs provide excellent robustness to both steering vector mismatch and rank over-determination. A detailed computational analysis has shown that out of the different Krylov-RDRCBs, the Conjugate Gradient RDRCB (CG-RDRCB) is the least computationally complex and, on the large array considered, is two orders of magnitude less complex than the standard RCB. We have also examined the efficient implementation of stopping-criterion based rank-selection rules in the Krylov-RDRCBs and found that they are useful for preventing numerical instability and can provide some further small performance enhancements. We have proposed two new low complexity stopping rules for use with the CG-RDRCB

that can be implemented at lower complexity than existing rules. On large arrays operating in non-stationary environments, it is often unfeasible to implement current full-dimension RCBs, due to the limited sample support and high complexity required. The proposed Krylov-RDRCBs can be implemented in these situations, as they are significantly less complex, converge faster and are amenable to parallel implementations.

IX. Acknowledgments

The authors would like to thank Dr. Nicholas Goddard and Dr. Les Hart for their continued support.

References

[1] J. R. Guerci, J. S. Goldstein, and I. S. Reed, "Optimal and Adaptive Reduced-Rank STAP," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 36, no. 2, pp. 647–663, Apr. 2000.

[2] S. D. Somasundaram, "Reduced Dimension Robust Capon Beamforming for Large Aperture Passive Sonar Arrays," *IET Radar, Sonar Navig.*, vol. 5, no. 7, pp. 707–715, Aug. 2011.

[3] S. P. Applebaum and D. J. Chapman, "Adaptive Arrays with Main Beam Constraints," *IEEE Trans. Antennas Propag.*, vol. AP-24, no. 5, pp. 650–662, Sep. 1976.

[4] M. H. Er and A. Cantoni, "Derivative Constraints for Broad-band Element Space Antenna Array Processors," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. ASSP-31, no. 6, pp. 1378–1393, Dec. 1983.

[5] K. M. Buckley and L. J. Griffiths, "An Adaptive Generalized Sidelobe Canceller with Derivative Constraints," *IEEE Trans. Antennas Propag.*, vol. AP-34, pp. 311–319, Mar. 1986.

[6] S. Zhang and I. L. Thng, "Robust Presteering Derivative Constraints for Broadband Antenna Arrays," *IEEE Trans. Signal Process.*, vol. 50, no. 1, pp. 1–10, Jan. 2002.

[7] H. Cox, R. M. Zeskind, and M. M. Owen, "Robust Adaptive Beamforming," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. ASSP-35, no. 10, pp. 1365–1376, Oct. 1987.

[8] 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.

[9] S. A. Vorobyov, A. B. Gershman, and Z.-Q. 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.

[10] P. Stoica, Z. Wang, and J. Li, "Robust Capon Beamforming," *IEEE Sig. Process. Lett.*, vol. 10, no. 6, pp. 172–175, Jun. 2003.

[11] J. Li, P. Stoica, and Z. Wang, "On Robust Capon Beamforming and Diagonal Loading," *IEEE Trans. Signal Process.*, vol. 51, no. 7, pp. 1702–1715, Jul. 2003.

[12] R. G. Lorenz and S. P. Boyd, "Robust Minimum Variance Beamforming," *IEEE Trans. Signal Process.*, vol. 53, no. 5, pp. 1684–1696, May 2005.

[13] J. Li and P. Stoica, *Robust Adaptive Beamforming*. New York: Wiley, 2005.

[14] J. Li, P. Stoica, and Z. Wang, "Doubly Constrained Robust Capon Beamformer," *IEEE Trans. Signal Process.*, vol. 52, no. 9, pp. 2407–2423, Sep. 2004.

[15] A. Beck and Y. C. Eldar, “Doubly Constrained Robust Capon Beamformer with Ellipsoidal Uncertainty Sets,” *IEEE Trans. Signal Process.*, vol. 55, no. 2, pp. 753–758, Feb. 2007.

[16] S. E. Nai, W. Ser, Z. L. Yu, and H. Chen, “Iterative Robust Minimum Variance Beamforming,” *IEEE Trans. Signal Process.*, vol. 59, no. 4, pp. 1601–1611, Apr. 2011.

[17] J. P. Lie, W. Ser, and C. M. S. See, “Adaptive Uncertainty Based Iterative Robust Capon Beamformer Using Steering Vector Mismatch Estimation,” *IEEE Trans. Signal Process.*, vol. 59, no. 9, pp. 4483–4488, Sep. 2011.

[18] M. Rubsamen and A. B. Gershman, “Robust Adaptive Beamforming Using Multidimensional Covariance Fitting,” *IEEE Trans. Signal Process.*, vol. 60, no. 2, pp. 740–753, Feb. 2012.

[19] S. D. Somasundaram, “Linearly Constrained Robust Capon Beamforming,” *IEEE Trans. Signal Process.*, vol. 60, no. 11, pp. 5845–5856, Nov. 2012.

[20] M. Rubsamen and M. Pesavento, “Maximally Robust Capon Beamformer,” to appear in *IEEE Trans. Signal Process.*

[21] S. D. Somasundaram, “Wideband Robust Capon Beamforming for Passive Sonar,” *IEEE J. Ocean. Eng.*, vol. 38, no. 2, pp. 308–322, Apr. 2013.

[22] T. Bowles, A. Jakobsson, and J. Chambers, “Detection of Cell-Cyclic Elements in Missampled Gene Expression Data Using a Robust Capon Estimator,” in *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing*, vol. 5, May 2004.

[23] S. V. der Tol and A.-J. V. der Veen, “Application of Robust Capon Beamforming to Radio Astronomical Imaging,” in *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing*, vol. 4, Mar. 4-6, 2005, pp. 1089–1092.

[24] A. Jakobsson and F. Gini, “Robust Estimation of Radar Reflectivities in Multibaseline InSAR,” *IEEE Trans. Aerosp. Electron. Syst.*, vol. 41, no. 2, pp. 751–758, Apr. 2005.

[25] Y. Wang, X. Li, Y. Sun, J. Li, and P. Stoica, “Adaptive Imaging for Forward-Looking Ground Penetrating Radar,” *IEEE Trans. Aerosp. Electron. Syst.*, vol. 41, no. 3, pp. 922–936, Jul. 2005.

[26] Y. Xie, B. Guo, L. Xu, J. Li, and P. Stoica, “Multistatic Adaptive Microwave Imaging for Early Breast Cancer Detection,” *IEEE Trans. Biomed. Eng*, vol. 53, no. 8, pp. 1647–1657, Aug. 2006.

[27] M.-R. Azimi-Sadjadi and N. Roseveare, “Wideband DOA Estimation Algorithms for Multiple Moving Sources using Unattended Acoustic Sensors,” *IEEE Trans. Aerosp. Electron. Syst.*, vol. 44, no. 4, pp. 1585–1599, Oct. 2008.

[28] L. Xu, J. Li, and P. Stoica, “Target Detection and Parameter Estimation for MIMO Radar Systems,” *IEEE Trans. Aerosp. Electron. Syst.*, vol. 44, no. 3, pp. 927–939, Jul 2008.

[29] S. D. Somasundaram and N. H. Parsons, “Evaluation of Robust Capon Beamforming for Passive Sonar,” *IEEE J. Ocean. Eng.*, vol. 36, no. 4, pp. 686–695, Oct. 2011.

[30] A. Hassanien, S. A. Vorobyov, and K.-M. Wong, “Robust Adaptive Beamforming Using Sequential Quadratic Programming: An Iterative Solution to the Mismatch Problem,” *IEEE Signal Process. Lett.*, vol. 15, pp. 733–736, 2008.

[31] A. Khabbazibasmenj, S. A. Vorobyov, and A. Hassanien, “Robust Adaptive Beamforming Based on Steering Vector Estimation With as Little as Possible Prior Information,” *IEEE Trans. Signal Process.*, vol. 60, no. 6, pp. 2974–2987, Jun. 2012.

[32] Y. Gu and A. Leshem, “Robust Adaptive Beamforming Based on Interference Covariance Matrix Reconstruction and Steering Vector Estimation,” *IEEE Trans. Signal Process.*, vol. 60, no. 7, pp. 3881–3885, Jul. 2012.

[33] B. Liao, S.-C. Chan, and K.-M. Tsui, “Recursive Steering Vector Estimation and Adaptive Beamforming under Uncertainties,” *IEEE Trans. Aerosp. Electron. Syst.*, vol. 49, no. 1, pp. 489–501, Jan. 2013.

[34] N. L. Owsley, "Systolic Array Adaptive Beamforming," in *Naval Underwater Systems Center Technical Report, NUSC 7981*, Apr. 1987.

[35] D. N. Swingler, "A Low-Complexity MVDR Beamformer for Use with Short Observation Times," *IEEE Trans. Signal Process.*, vol. 47, no. 4, pp. 1154–1160, Apr. 1999.

[36] H. Cox and H. Lai, "Sub-Aperture Beam-Based Adaptive Beamforming for Large Dynamic Arrays," in *Proc. 38th Asilomar Conf. Signals, Systems, Comput.*, vol. 2, Nov. 2004, pp. 2355–2358.

[37] Y. Doisy, L. Deruaz, and R. Been, "Interference Suppression of Subarray Adaptive Beamforming in Presence of Sensor Dispersions," *IEEE Trans. Signal Process.*, vol. 58, no. 8, pp. 4195–4212, Aug. 2010.

[38] D. A. Gray, "Formulation of the Maximum Signal-to-Noise Ratio Array Processor in Beam Space," *J. Acoust. Soc. Am.*, vol. 72, no. 4, pp. 1195–1201, Oct. 1982.

[39] G. Bienvenu and L. Kopp, "Decreasing High Resolution Method Sensitivity by Conventional Beamformer Preprocessing," in *IEEE International Conference on Acoustics, Speech and Signal Processing*, vol. 9, Honolulu, HI, Mar. 1984, pp. 714–717.

[40] H. L. Van Trees, *Optimum Array Processing: Part IV of Detection, Estimation and Modulation Theory*. New York: Wiley, 2002.

[41] P. Forster and G. Vezzosi, "Application of Spheroidal Sequences to Array Processing," in *Proc. ICASSP '87*, Dallas, TX, Apr. 1987, pp. 2268–2271.

[42] A. M. Haimovich and Y. Bar-Ness, "An Eigenanalysis Interference Canceler," *IEEE Trans. Signal Process.*, vol. 39, no. 1, pp. 76–84, Jan. 1991.

[43] J. S. Goldstein and I. S. Reed, "Reduced-Rank Adaptive Filtering," *IEEE Trans. Signal Process.*, vol. 45, no. 2, pp. 492–496, Feb. 1997.

[44] ——, "Subspace Selection for Partially Adaptive Sensor Array Processing," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 33, no. 2, pp. 539–544, Apr. 1997.

[45] C. H. Gierull, "Statistical Analysis of the Eigenvector Projection Method for Adaptive Spatial Filtering of Interference," *IEE Proceedings on Radar, Sonar and Navigation*, vol. 144, no. 2, pp. 57–63, Apr. 1997.

[46] G. Dietl, M. D. Zoltowski, and M. Joham, "Recursive Reduced-Rank Adaptive Equalization for Wireless Communications," in *Proc. SPIE*, vol. 4395, Apr. 2001.

[47] ——, "Reduced-Rank Equalization for EDGE via Conjugate Gradient Implementation of Multi-Stage Nested Wiener Filter," in *IEEE Vehicular Technology Conference*, vol. 3, Oct. 2001, pp. 1912–1916.

[48] G. Dietl, "Conjugate Gradient Implementation of Multi-Stage Nested Wiener Filter for Reduced-Dimension Processing," in *MSc. Dissertation, Munich University of Technology*, Munich, Germany, May 2001.

[49] S. Burykh and K. Abed-Meraim, "Reduced-Rank Adaptive Filtering Using Krylov-Subspace," *EURASIP J. Appl. Signal Process.*, vol. 2002, no. 1, pp. 1387–1400, Jan. 2002.

[50] M. E. Weippert, J. D. Hiemstra, J. S. Goldstein, and M. D. Zoltowski, "Insights from the Relationship Between the Multistage Wiener Filter and the Method of Conjugate Gradients," in *Proc. IEEE Workshop on Sens. Array Multichannel Signal Processing*, Aug. 4-6, 2002, pp. 388–392.

[51] M. L. Honig and W. Xiao, "Performance of Reduced-Rank Linear Interference Suppression," *IEEE Trans. Information Theory*, vol. 47, no. 5, pp. 1928–1946, Jul. 2001.

[52] M. L. Honig and J. S. Goldstein, "Adaptive Reduced-Rank Interference Suppression Based on the Multistage Wiener Filter," *IEEE Transactions on Communications*, vol. 50, no. 6, pp. 986–994, Jun. 2002.

[53] G. H. Golub and C. F. V. Loan, *Matrix Computations*, 3^{rd} ed. The John Hopkins University Press, 1996.

[54] L. L. Scharf, E. K. P. Chong, M. D. Zoltowski, J. S. Goldstein, and I. S. Reed, "Subspace Expansion and the Equivalence of Conjugate Direction Multistage Wiener Filters," *IEEE Trans. Signal Process.*, vol. 56, no. 10, pp. 5013–5019, Oct. 2008.

[55] 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.

[56] D. A. Pados and S. N. Batalama, "Joint Space-Time Auxiliary Vector Filtering for DS/CDMA Systems with Antenna Arrays," *IEEE Trans. Commun.*, vol. 47, no. 9, pp. 1406–1415, Sept. 1999.

[57] 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.

[58] M. R. Hestenes and E. Stiefel, "Methods of Conjugate Gradients for Solving Linear Systems," *Journal of Research of the National Bureau of Standards*, vol. 49, no. 6, pp. 409–436, Dec. 1952.

[59] G. K. Boray and M. D. Srinath, "Conjugate Gradient Techniques for Adaptive Filtering," *IEEE Trans. Circuits and Systems-1*, vol. 39, no. 1, pp. 1–10, Jan. 1992.

[60] P. S. Chang and A. N. Willson, "Analysis of Conjugate Gradient Algorithms for Adaptive Filtering," *IEEE Trans. Signal Process.*, vol. 48, no. 2, pp. 409–418, Feb. 2000.

[61] H. Ge, L. L. Scharf, and M. Lundberg, "Reduced-Rank Multiuser Detectors Based on Vector and Matrix Conjugate Gradient Wiener Filters," in *IEEE 5th Workshop on Signal Processing Advances in Wireless Communications*, Lisboa,Portugal, Jul. 2004, pp. 189–193.

[62] H. Ge, M. Lundberg, and L. L. Scharf, "Warp Convergence in Conjugate Gradient Wiener Filters," in *IEEE Sensor Array and Multichannel Signal Processing Workshop*, Barcelona,Spain, Jul. 2004, pp. 109–113.

[63] H. Ge, I. P. Kirsteins, and L. L. Scharf, "Data Dimension Reduction Using Krylov Subspaces: Making Adaptive Beamformers Robust to Model Order-Determination," in *31^{st} IEEE International Conference on Acoustics, Speech and Signal Processing*, vol. 4, Toulouse, Fr, May. 2006, pp. 1001–1004.

[64] I. P. Kirsteins and H. Ge, "Performance Analysis of Krylov Space Adaptive Beamformers," in *4^{th} IEEE Workshop on Sensor Array and Multichannel Processing*, Waltham, MA, Jul. 2006, pp. 16–20.

[65] R. C. de Lamare, M. Haardt, and R. Sampaio-Neto, "Blind Adaptive Constrained Reduced-Rank Parameter Estimation Based on Constant Modulus Design for CDMA Interference Suppression," *IEEE Trans. Signal Process.*, vol. 56, no. 6, pp. 2470–2482, Jun. 2008.

[66] M. Yukawa, R. C. de Lamare, and I. Yamada, "Robust Reduced-Rank Adaptive Algorithm Based on Parallel Subgradient Projection and Krylov Subspace," *IEEE Trans. Signal Process.*, vol. 57, no. 12, pp. 4660–4674, Dec. 2009.

[67] L. Wang and R. C. de Lamare, "Constrained adaptive filtering algorithms based on conjugate gradient techniques for beamforming," *IET Signal Processing*, vol. 4, pp. 686–697, 2010.

[68] 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.

[69] A. Hassanien and S. A. Vorobyov, "A Robust Adaptive Dimension Reduction Technique With Application to Array Processing," *IEEE Signal Process. Lett.*, vol. 16, no. 1, pp. 22–25, Jan. 2009.

[70] S. D. Somasundaram, "A Framework for Reduced Dimension Robust Capon Beamforming," in *Proc. IEEE Workshop on Statistical Signal Processing*, Nice, France, Jun. 28-30 2011, pp. 425–428.

[71] M. Yukawa, "Rank-Selection for Krylov-Subspace Based Adaptive Filtering Techniques," in *Proc. 2011 IEEE International Conference on Information, Communications and Signal Processing*, Singapore, Dec. 2011, pp. 1–4.

[72] S. D. Somasundaram, N. H. Parsons, P. Li, and R. C. de Lamare, "Data-Adaptive Reduced-Dimension Robust Capon Beamforming," in *Proc. 2013 IEEE International Conference on Acoustics, Speech and Signal Processing*, Vancouver, CA, May 26-31 2013, pp. 4159–4163.

[73] W. H. Press, S. A. Teukolsky, W. T. Vetterling, and B. P. Flannery, *Numerical Reciped in C: The Art of Scientific Computing*, 2^{nd} ed. Cambridge, U.K.: Cambridge University Press, 1992.

[74] S. D. Somasundaram, A. Jakobsson, and N. H. Parsons, "Robust and Automatic Data-Adaptive Beamforming for Multi-Dimensional Arrays," *IEEE Trans. Geosci. Remote Sens.*, vol. 50, no. 11, pp. 4642–4656, Nov. 2012.

[75] J. R. Shewchuk, "An Introduction to the Conjugate Gradient Method Without the Agonizing Pain," in Technical Report, *Carnegie Mellon University, Pittsburgh, PA*, 1994.

[76] G. Strang, *Linear Algebra and its Applications (3rd Edition)*. Harcourt Brace Jovanovich, 1988.

[77] P. Stoica and R. L. Moses, *Spectral Analysis of Signals*. Prentice Hall, 2005.

[78] S. L. Marple, *Digital Spectral Analysis with Applications*. Prentice-Hall, 1987.