Reduced-Rank Space–Time Adaptive Interference Suppression With Joint Iterative Least Squares Algorithms for Spread-Spectrum Systems

Rodrigo C. de Lamare, *Member, IEEE*, and Raimundo Sampaio-Neto

***Abstract*—This paper presents novel adaptive space–time reduced-rank interference-suppression least squares (LS) algorithms based on a joint iterative optimization of parameter vectors. The proposed space–time reduced-rank scheme consists of a joint iterative optimization of a projection matrix that performs dimensionality reduction and an adaptive reduced-rank parameter vector that yields the symbol estimates. The proposed techniques do not require singular value decomposition (SVD) and automatically find the best set of basis for reduced-rank processing. We present LS expressions for the design of the projection matrix and the reduced-rank parameter vector, and we conduct an analysis of the convergence properties of the LS algorithms. We then develop recursive LS (RLS) adaptive algorithms for their computationally efficient estimation and an algorithm that automatically adjusts the rank of the proposed scheme. A convexity analysis of the LS algorithms is carried out along with the development of a proof of convergence for the proposed algorithms. Simulations for a space–time interference suppression application with a direct-sequence code-division multiple-access (DS-CDMA) system show that the proposed scheme outperforms in convergence and tracking the state-of-the-art reduced-rank schemes at a comparable complexity.**

***Index Terms*—Interference suppression, iterative methods, least squares (LS) algorithms, space–time adaptive processing (STAP), spread-spectrum systems.**

I. Introduction

SPACE–TIME adaptive processing (STAP) techniques have become a fundamental enabling technology of modern systems encountered in communications [1], radar and sonar [2], [3], and navigation [4]. The basic idea is to gather data samples from an antenna array and process them both spatially and temporally via a linear combination of adaptive weights. In particular, STAP algorithms have found numerous applications in modern wireless communications based on spread-spectrum systems and code-division multiple access (CDMA) [5], [6]. These systems implemented with direct-sequence (DS) signaling are found in third-generation cellular telephony [7]–[9], indoor wireless networks [10], satellite communications, and ultrawideband technology [11] and are being considered for future systems with multicarrier (MC) versions such as MC-CDMA and MC-DS-CDMA [12] and in conjunction with multiple antennas [13]. The advantages of spread-spectrum systems include good performance in multipath channels, flexibility in the allocation of channels, increased capacity in bursty and fading environments, and the ability to share bandwidth with narrow-band communication systems without performance degradation [5].

There are numerous algorithms with different tradeoffs between performance and complexity for designing STAP techniques [14]. Among them, least squares (LS)-based algorithms are often the preferred choice with respect to convergence performance. However, when the number of filter elements in the STAP algorithm is large, they require a large number of samples to reach its steady-state behavior and may encounter problems in tracking the desired signal. Reduced-rank STAP techniques [15]–[40] are powerful and effective approaches in low-sample support situations and in problems with large filters. These algorithms can effectively exploit the low-rank nature of signals that are found in spread-spectrum communications. Their advantages are faster convergence speed and better tracking performance than full-rank techniques when dealing with a large number of weights. It is well known that the optimal reduced-rank approach is based on the singular value decomposition (SVD) of the known input data covariance matrix $\mathbf{R}$ [15]. However, this covariance matrix must be estimated. The approach taken to estimate $\mathbf{R}$ and perform dimensionality reduction is of central importance and plays a key role in the performance of the system. Numerous reduced-rank strategies have been proposed in the last two decades. Among the first methods are those based on the SVD of time-averaged estimates of $\mathbf{R}$ [15]–[20], in which the dimensionality reduction is carried out by a projection matrix formed by appropriately selected eigenvectors that are computed with the SVD. An effective approach to address the problem of selection of eigenvectors, which is known as the cross-spectral method and results in improved performance, was considered in [21]. Iterative algorithms that avoid the SVD but do not fully exploit the structure of the data for reduced-rank processing were reported in [22] and [23]. A more recent and elegant approach to the problem was taken with the advent of the multistage Wiener filter (MSWF) [24], which was later extended

Manuscript received January 24, 2009; revised July 8, 2009 and October 9, 2009. First published December 15, 2009; current version published March 19, 2010. This work was supported in part by the Ministry of Defence (MoD), U.K., Project MoD, No. RT/COM/S/021, and in part by the National Council for Research and Development (CNPQ). The review of this paper was coordinated by Prof. H. H. Nguyen.

R. C. de Lamare is with the Communications Research Group, Department of Electronics, University of York, Y010 5DD York, U.K. (e-mail: rcdl500@ohm.york.ac.uk).

R. Sampaio-Neto is with the Center for Studies in Telecommunications/Pontifical Catholic University of Rio de Janeiro, 22453-900 Rio de Janeiro, Brazil (e-mail: raimundo@cetuc.puc-rio.br).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TVT.2009.2038391

to adaptive versions by Honig and Goldstein [25], STAP applications [26], and other related techniques [27]. Another method that was reported about the same time as the MSWF is the auxiliary vector filtering (AVF) algorithm [28]–[32]. A reduced-rank method based on interpolated filters with time-varying interpolators was reported in [34]–[36] for temporal processing, and an associated STAP version was considered in [37]; however, this approach shows significant performance degradation with small ranks. A key limitation with the existing reduced-rank STAP techniques is the lack or a deficiency with the exchange of information between the projection matrix that carries out dimensionality reduction and the subsequent reduced-rank filtering.

In this paper, we propose reduced-rank STAP LS algorithms for interference suppression in spread-spectrum systems. The proposed algorithms require neither SVD nor prior knowledge of the reduced model order. The proposed reduced-rank STAP scheme consists of a joint iterative optimization of a projection matrix that performs dimensionality reduction and is followed by an adaptive reduced-rank filter. The key aspect of the proposed approach is to exchange information between the tasks of dimensionality reduction and reduced-rank processing. The proposed STAP scheme builds on the temporal scheme first reported in [38] with stochastic gradient algorithms and extends it to the case of spatiotemporal processing and to a deterministic exponentially weighted LS design criterion. We develop LS optimization algorithms and expressions for the joint design of the projection matrix and the reduced-rank filter. We derive recursive LS (RLS) adaptive algorithms for their computationally efficient implementation along with a complexity study of the proposed and existing algorithms. We also devise an algorithm that automatically adjusts the rank of the filters utilized in the proposed STAP scheme. A convexity analysis of the proposed LS optimization of the filters is conducted, and an analysis of the convergence of the proposed RLS algorithms is also carried out. The performance of the proposed scheme is assessed via simulations for a space–time interference suppression application in DS-CDMA systems. The main contributions of this paper are summarized as follows: 1) a reduced-rank STAP scheme for spatiotemporal processing of signals; 2) LS expressions and recursive algorithms for STAP parameter estimation; 3) an algorithm for automatically adjusting the rank of the filters; and 4) convexity analysis and convergence proof of the proposed LS-based algorithms.

This paper is organized as follows. Section II presents the space–time system model, and Section III states the reduced-rank estimation problem. Section IV presents the novel reduced-rank scheme, the joint iterative optimization, and the LS design of the filters. Section V derives the RLS and the rank adaptation algorithms for implementing the proposed scheme. Section VI develops the analysis of the proposed algorithms. Section VII shows and discusses the simulations, whereas Section VIII gives the conclusions.

II. Space–Time System Model

We consider the uplink of the DS-CDMA system with symbol interval T, chip period T_c, spreading gain $N = T/Tc$, K users, multipath channels with L propagation paths, and $L < N$. The system is equipped with an antenna that consists of a uniform linear array (ULA) and J sensor elements [2], [3]. In the model adopted, the intersymbol interference span and contribution are functions of the processing gain N and L [6]. For instance, we assume that $L \leq N$, which results in the interference between three symbols in total—the current one, the previous, and the successive symbols. The spacing between the ULA elements is $d = \lambda_c/2$, where λ_c is the carrier wavelength. We assume that the channel is constant during each symbol, the base station receiver is perfectly synchronized, and the delays of the propagation paths are multiples of the chip rate. The received signal after filtering by a chip-pulse matched filter and sampled at the chip period yields the $JM \times 1$ received vector at time i, i.e.,

$$\boldsymbol{r}[i] = \sum_{k=1}^{K} A_k b_k[i-1]\bar{\boldsymbol{p}}_k[i-1] + A_k b_k[i]\boldsymbol{p}_k[i] + A_k b_k[i+1]\tilde{\boldsymbol{p}}_k[i+1] + \boldsymbol{n}[i] \quad (1)$$

where $M = N + L - 1$, the complex Gaussian noise vector is $\boldsymbol{n}[i] = [n_1[i] \ \ldots \ n_{JM}[i]]^T$ with $E[\boldsymbol{n}[i]\boldsymbol{n}^H[i]] = \sigma^2\boldsymbol{I}$, $(\cdot)^T$ and $(\cdot)^H$ denote the transpose and the Hermitian transpose, respectively, and $E[\cdot]$ stands for the expected value. The spatial signatures for previous, current, and future data symbols are

$$\begin{aligned} \bar{\boldsymbol{p}}_k[i-1] &= \bar{\boldsymbol{\mathcal{F}}}_k \boldsymbol{\mathcal{H}}_k[i-1] \\ \boldsymbol{p}_k[i] &= \boldsymbol{\mathcal{F}}_k \boldsymbol{\mathcal{H}}_k[i] \\ \tilde{\boldsymbol{p}}_k[i+1] &= \tilde{\boldsymbol{\mathcal{F}}}_k \boldsymbol{\mathcal{H}}_k[i+1] \end{aligned} \quad (2)$$

where $\bar{\boldsymbol{\mathcal{F}}}_k$, $\boldsymbol{\mathcal{F}}_k$, and $\tilde{\boldsymbol{\mathcal{F}}}_k$ are block diagonal matrices with versions of segments of the signature sequence $\boldsymbol{s}_k = [a_k(1) \ \ldots \ a_k(N)]^T$ of user k shifted down by one position (one chip) and given by $\bar{\boldsymbol{\mathcal{F}}}_k = \mathrm{diag}(\bar{\boldsymbol{\mathcal{C}}}_k, \bar{\boldsymbol{\mathcal{C}}}_k, \ldots, \bar{\boldsymbol{\mathcal{C}}}_k)$, $\boldsymbol{\mathcal{F}}_k = \mathrm{diag}(\boldsymbol{\mathcal{C}}_k, \boldsymbol{\mathcal{C}}_k, \ldots, \boldsymbol{\mathcal{C}}_k)$, and $\tilde{\boldsymbol{\mathcal{F}}}_k = \mathrm{diag}(\tilde{\boldsymbol{\mathcal{C}}}_k, \tilde{\boldsymbol{\mathcal{C}}}_k, \ldots, \tilde{\boldsymbol{\mathcal{C}}}_k)$. The structure of the $M \times L$ matrices $\bar{\boldsymbol{\mathcal{C}}}_k$, $\boldsymbol{\mathcal{C}}_k$, and $\tilde{\boldsymbol{\mathcal{C}}}_k$ is detailed in [9]. The $JL \times 1$ space–time channel vector is given by

$$\boldsymbol{\mathcal{H}}_{\boldsymbol{k}}[i] = \left[\boldsymbol{h}_{k,0}^T[i] | \, \boldsymbol{h}_{k,1}^T[i] | \ \ldots \ | \boldsymbol{h}_{k,J-1}^T[i]\right]^T \quad (3)$$

where $\boldsymbol{h}_{k,l}[i] = [h_{k,0}^{(l)}[i] \ \ldots \ h_{k,L-1}^{(l)}[i]]^T$ is the $L \times 1$ channel vector of user k at antenna element l with their associated directions of arrival (DoAs) $\phi_{k,m}$. The DoAs are assumed different for each user and path [24].

III. Reduced-Rank Space–Time Adaptive Processing for Interference Suppression and Problem Statement

Here, we outline the main problem of the STAP design for interference suppression in spread-spectrum systems, and we consider the design of reduced-rank STAP algorithms using an LS approach. The main goal of the STAP algorithms is to jointly perform temporal filtering with spatial filtering (beamforming) through adaptive combination of filter coefficients.

Let us consider the space–time received signals of Section II and the data organized in $JM \times 1$ vectors $\boldsymbol{r}[i]$. To process this data vector, one can design a STAP algorithm that consists of a

$JM \times 1$ filter $\boldsymbol{w}[i] = [w_1^{[i]} w_2^{[i]} \ \dots \ w_{JM}^{[i]}]^T$, which adaptively and linearly combines its coefficients with the received data samples to yield an estimate $x[i] = \boldsymbol{w}^H[i]\boldsymbol{r}[i]$. The design of $\boldsymbol{w}[i]$ can be performed via the minimization of the exponentially weighted LS cost function

$$\mathcal{C}\left(\boldsymbol{w}[i]\right) = \sum_{l=1}^{i} \lambda^{i-l} \left| d[l] - \boldsymbol{w}^H[i]\boldsymbol{r}[l] \right|^2 \tag{4}$$

where $d[l]$ is the desired signal, and λ stands for the forgetting factor. Solving for $\boldsymbol{w}[i]$, we obtain

$$\boldsymbol{w}[i] = \boldsymbol{R}^{-1}[i]\boldsymbol{p}[i] \tag{5}$$

where $\boldsymbol{R}[i] = \sum_{l=1}^{i} \lambda^{i-l}\boldsymbol{r}[l]\boldsymbol{r}^H[l]$ is the time-averaged correlation matrix, and $\boldsymbol{p}[i] = \sum_{l=1}^{i} \lambda^{i-l} d^*[l]\boldsymbol{r}[l]]$ is the cross-correlation vector.

A problem with STAP algorithms is that the laws that govern their convergence and tracking behavior imply that the performance is a function of JM, which is the number of elements in the filter. Thus, large JM implies slow convergence and poor tracking performance. A reduced-rank STAP algorithm attempts to circumvent this limitation by exploiting the low-rank nature of spread-spectrum systems and performing dimensionality reduction. This dimensionality reduction reduces the number of adaptive coefficients and extracts the key features of the processed data. It is accomplished by projecting the received vectors onto a lower dimensional subspace. Specifically, consider a $JM \times D$ projection matrix $\boldsymbol{T}_D[i]$ that carries out a dimensionality reduction on the received data as given by

$$\bar{\boldsymbol{r}}[i] = \boldsymbol{T}_D^H[i]\boldsymbol{r}[i] \tag{6}$$

where, in what follows, all D-dimensional quantities are denoted with a "bar." The resulting projected received vector $\bar{\boldsymbol{r}}[i]$ is the input to a tapped-delay line represented by the D vector $\bar{\boldsymbol{w}}[i] = [\bar{w}_1^{[i]} \ \bar{w}_2^{[i]} \ \dots \ \bar{w}_D^{[i]}]^T$. The reduced-rank STAP output is

$$x[i] = \bar{\boldsymbol{w}}^H[i]\bar{\boldsymbol{r}}[i]. \tag{7}$$

If we consider the LS design in (4) with the reduced-rank parameters, we obtain

$$\bar{\boldsymbol{w}}[i] = \bar{\boldsymbol{R}}^{-1}[i]\bar{\boldsymbol{p}}[i] \tag{8}$$

where $\bar{\boldsymbol{R}}[i] = \sum_{l=1}^{i} \lambda^{i-l}\bar{\boldsymbol{r}}[l]\bar{\boldsymbol{r}}^H[l] = \boldsymbol{T}_D^H[i]\boldsymbol{R}[i]\boldsymbol{T}_D[i]$ is the reduced-rank correlation matrix, and $\bar{\boldsymbol{p}}[i] = \sum_{l=1}^{i} \lambda^{i-l} d^*[l]\bar{\boldsymbol{r}}[l] = \boldsymbol{T}_D^H[i]\boldsymbol{p}[i]$ is the cross-correlation vector of the reduced-rank model. The associated sum of error squares (SES) for a rank-D STAP is expressed by

$$\begin{aligned} \text{SES} &= \sigma_d^2 - \bar{\boldsymbol{p}}^H[i]\bar{\boldsymbol{R}}^{-1}[i]\bar{\boldsymbol{p}}[i] \\ &= \sigma_d^2 - \boldsymbol{p}^H[i]\boldsymbol{T}_D[i]\left(\boldsymbol{T}_D^H[i]\boldsymbol{R}[i]\boldsymbol{T}_D[i]\right)^{-1}\boldsymbol{T}_D^H[i]\boldsymbol{p}[i] \end{aligned} \tag{9}$$

where $\sigma_d^2 = \sum_{l=1}^{i} \lambda^{i-l}|d(l)|^2$. The development above shows us that the key aspect for constructing reduced-rank STAP schemes is the design of $\boldsymbol{T}_D[i]$ since the SES in (9) depends on $\boldsymbol{p}[i]$, $\boldsymbol{R}[i]$, and $\boldsymbol{T}_D[i]$. The quantities $\boldsymbol{p}[i]$ and $\boldsymbol{R}[i]$ are common to both reduced-rank and full-rank STAP designs; however, the projection matrix $\boldsymbol{T}_D[i]$ plays a key role in the dimensionality reduction and in the performance. The strategy is to find the most appropriate tradeoff between the model bias and variance [15] by adjusting the rank D and exchanging information between $\boldsymbol{T}_D[i]$ and $\boldsymbol{w}[i]$. For instance, numerically evaluating the SES expression in (9), one can verify the convergence and steady-state performance of reduced-rank STAP algorithms. Next, we present the proposed reduced-rank STAP approach.

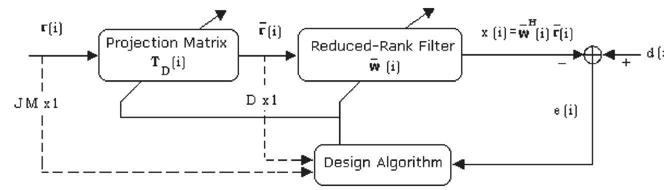

Fig. 1. Proposed reduced-rank STAP scheme.

IV. Proposed Reduced-Rank STAP and Least Squares Design

Here, we detail the principles of the proposed reduced-rank STAP scheme and present an LS design approach for the filters. The proposed reduced-rank STAP scheme is depicted in Fig. 1 and is formed by a projection matrix $\boldsymbol{T}_D[i]$ with dimensions $JM \times D$ that is responsible for the dimensionality reduction and a $D \times 1$ reduced-rank filter $\bar{\boldsymbol{w}}[i]$. The $JM \times 1$ received data vector $\boldsymbol{r}[i]$ is mapped by $\boldsymbol{T}_D[i]$ into a $D \times 1$ reduced-rank data vector $\bar{\boldsymbol{r}}[i]$. The reduced-rank filter $\bar{\boldsymbol{w}}[i]$ linearly combines $\bar{\boldsymbol{r}}[i]$ to yield a scalar estimate $x[i]$. The key strategy of the proposed framework lies in the joint design of the projection matrix $\boldsymbol{T}_D[i]$ and the reduced-rank filter $\bar{\boldsymbol{w}}[i]$ according to the LS criterion. The exchange of information between $\boldsymbol{T}_D[i]$ and $\bar{\boldsymbol{w}}[i]$ is different from the MSWF [24]–[26] and the AVF techniques [28]–[32]. In particular, the expressions of the filters that are obtained for the proposed reduced-rank STAP scheme allow a more efficient introduction of the bias than that of the MSWF and the AVF by alternating the recursions for $\boldsymbol{T}_D[i]$ and $\bar{\boldsymbol{w}}[i]$. In addition, the proposed STAP scheme is based on a subspace projection designed according to a joint and iterative minimization of the LS cost function and achieves better performance than the Krylov subspace of the MSWF and the AVF.

Let us now detail the quantities that are involved in the proposed reduced-rank STAP scheme. Specifically, the projection matrix $\boldsymbol{T}_D[i]$ is structured as a bank of D full-rank filters with dimensions $JM \times 1$, which are described by

$$\boldsymbol{t}_d[i] = \left[t_{1,d}^{[i]} \ t_{2,d}^{[i]} \ \dots \ t_{JM,d}^{[i]}\right]^T, \qquad d = 1, \dots, D. \tag{10}$$

The filters $\boldsymbol{t}_d[i]$ are then gathered and organized, yielding

$$\boldsymbol{T}_D[i] = \left[\boldsymbol{t}_1^{[i]} \mid \boldsymbol{t}_2^{[i]} \mid \ \dots \ | \boldsymbol{t}_D^{[i]}\right]. \tag{11}$$

The output estimate $x[i]$ of the reduced-rank STAP scheme can be expressed as a function of the received data $\boldsymbol{r}[i]$, the projection matrix $\boldsymbol{T}_D[i]$, and the reduced-rank filter $\bar{\boldsymbol{w}}[i]$ as given by

$$x[i] = \bar{\boldsymbol{w}}^H[i]\boldsymbol{T}_D^H[i]\boldsymbol{r}[i] = \bar{\boldsymbol{w}}^H[i]\bar{\boldsymbol{r}}[i]. \tag{12}$$

Interestingly, for $D = 1$, the proposed STAP scheme becomes a conventional full-rank STAP algorithm with an additional weight parameter w_D that can be seen as a gain. For $D > 1$, the signal processing tasks are changed; moreover, the full-rank filters $\boldsymbol{t}_d[i]$ perform dimensionality reduction, and the reduced-rank filter estimates the desired signal.

To design the projection matrix $\boldsymbol{T}_D[i]$ and the reduced-rank filter $\bar{\boldsymbol{w}}[i]$, we need to adopt an appropriate design criterion. We will resort to an exponentially weighted LS approach since it is mathematically tractable and results in joint optimization algorithms that can track time-varying signals by adjusting the forgetting factor λ. The design of the proposed scheme amounts to solving the following optimization problem:

$$\left[\boldsymbol{T}_{D,\mathrm{opt}}[i], \bar{\boldsymbol{w}}^H_{\mathrm{opt}}[i]\right] = \arg \min_{\boldsymbol{T}_D[i], \bar{\boldsymbol{w}}^H[i]} \sum_{l=1}^{i} \lambda^{i-l} \left|d[l] - \bar{\boldsymbol{w}}^H[i]\boldsymbol{T}_D^H[i]\boldsymbol{r}[l]\right|^2 \Bigg]. \quad (13)$$

To solve the above minimization problem, let us then consider the cost function

$$\mathcal{C}\left(\boldsymbol{T}_D[i], \bar{\boldsymbol{w}}^H[i]\right) = \sum_{l=1}^{i} \lambda^{i-l} \left|d[l] - \bar{\boldsymbol{w}}^H[i]\boldsymbol{T}_D^H[i]\boldsymbol{r}[l]\right|^2 \Bigg]. \quad (14)$$

Minimizing (14) with respect to $\boldsymbol{T}_D[i]$, we obtain

$$\boldsymbol{T}_{D,\mathrm{opt}}[i] = \boldsymbol{R}^{-1}[i]\boldsymbol{P}_D[i]\boldsymbol{R}^{-1}_{\bar{\boldsymbol{w}}}[i] \quad (15)$$

where $\boldsymbol{P}_D[i] = \sum_{l=1}^{i} \lambda^{i-l} d^*[l]\boldsymbol{r}[l]\bar{\boldsymbol{w}}^H[i]$, the time-averaged correlation matrix is $\boldsymbol{R}[i] = \sum_{l=1}^{i} \lambda^{i-l}\boldsymbol{r}[l]\boldsymbol{r}^H[l]$, and $\boldsymbol{R}_{\bar{\boldsymbol{w}}}[i] = \bar{\boldsymbol{w}}[i]\bar{\boldsymbol{w}}^H[i]$. Note that we have opted for computing $\boldsymbol{R}_{\bar{\boldsymbol{w}}}[i]$ as $\boldsymbol{R}_{\bar{\boldsymbol{w}}}[i] = \sum_{l=1}^{i} \lambda^{i-l}\bar{\boldsymbol{w}}[l]\bar{\boldsymbol{w}}^H[l]$ with a regularization term introduced at the beginning of the iterations to allow the computation of its inverse. For this reason, the latter approach will be adopted for the derivation of adaptive algorithms. Minimizing (14) with respect to $\bar{\boldsymbol{w}}[i]$, the reduced-rank filter becomes

$$\bar{\boldsymbol{w}}_{\mathrm{opt}}[i] = \bar{\boldsymbol{R}}^{-1}[i]\bar{\boldsymbol{p}}[i] \quad (16)$$

where $\bar{\boldsymbol{p}}[i] = \boldsymbol{T}^H_{D,\mathrm{opt}}[i]\sum_{l=1}^{i} \lambda^{i-l} d^*[l]\boldsymbol{r}[l] = \sum_{l=1}^{i} \lambda^{i-l} d^*[l]\bar{\boldsymbol{r}}[l]]$, $\bar{\boldsymbol{R}}[i] = \boldsymbol{T}^H_{D,\mathrm{opt}}[i]\sum_{l=1}^{i} \lambda^{i-l}\boldsymbol{r}[l]\boldsymbol{r}^H[l]\boldsymbol{T}_{D,\mathrm{opt}}[i]$. The associated SES for the proposed reduced-rank STAP scheme is

$$\mathrm{SES} = \sigma_d^2 - \bar{\boldsymbol{p}}^H[i]\bar{\boldsymbol{R}}^{-1}[i]\bar{\boldsymbol{p}}[i] \quad (17)$$

where $\sigma_d^2 = \sum_{l=1}^{i} \lambda^{i-l}|d[l]|^2$. Note that the expressions in (15) and (16) are not closed-form solutions for $\bar{\boldsymbol{w}}_{\mathrm{opt}}[i]$ and $\boldsymbol{T}_{D,\mathrm{opt}}[i]$ since (15) is a function of $\bar{\boldsymbol{w}}_{\mathrm{opt}}[i]$, and (16) depends on $\boldsymbol{T}_{D,\mathrm{opt}}[i]$. Therefore, they have to be iterated with an initial guess to obtain a solution. The expressions in (15) and (16) require the inversion of matrices, which entails cubic complexity with JM and D. Computing the SES in (17), it can numerically verify the convergence and steady-state performance of reduced-rank STAP algorithms, namely, the proposed, the MSWF [25], and the AVF [32]. To reduce the complexity, we will derive RLS algorithms in Section V. The rank D must be set by the designer to ensure appropriate performance, or a mechanism for automatically adjusting the rank should be adopted. We will also present an automatic rank-selection algorithm in what follows.

V. Proposed Recursive Least Squares and Rank-Selection Algorithms

Here, we propose RLS algorithms that efficiently implement the LS design of Section IV and estimate the filters $\boldsymbol{T}_{D,\mathrm{opt}}[i]$ and $\bar{\boldsymbol{w}}_{\mathrm{opt}}[i]$ with the filters $\boldsymbol{T}_D[i]$ and $\bar{\boldsymbol{w}}[i]$, respectively. We also develop rank-selection algorithms that automatically adjust the rank D of the proposed STAP algorithm. An analysis of the computational requirements of the proposed and analyzed algorithms is also included.

A. *Proposed RLS Algorithm*

To derive an RLS algorithm for the proposed scheme, we consider (15) and derive a recursive procedure for computing the parameters of $\boldsymbol{T}_D[i]$. Let us define

$$\begin{aligned}\boldsymbol{P}[i] &= \boldsymbol{R}^{-1}[i]\\ \boldsymbol{Q}_{\bar{\boldsymbol{w}}}[i] &= \boldsymbol{R}^{-1}_{\bar{\boldsymbol{w}}}[i-1]\\ \boldsymbol{P}_D[i] &= \lambda\boldsymbol{P}_D[i-1] + d^*[i]\boldsymbol{r}[i]\bar{\boldsymbol{w}}^H[i]\end{aligned} \quad (18)$$

and rewrite the expression in (15) as follows:

$$\begin{aligned}\boldsymbol{T}_D[i] &= \boldsymbol{P}[i]\boldsymbol{P}_D[i]\boldsymbol{Q}_{\bar{\boldsymbol{w}}}[i]\\ &= \lambda\boldsymbol{P}[i]\boldsymbol{P}_D[i-1]\boldsymbol{Q}_{\bar{\boldsymbol{w}}}[i] + d^*[i]\boldsymbol{P}[i]\boldsymbol{r}[i]\bar{\boldsymbol{w}}^H[i]\boldsymbol{Q}_{\bar{\boldsymbol{w}}}[i]\\ &= \boldsymbol{T}_D[i-1] - \boldsymbol{k}[i]\boldsymbol{P}[i-1]\boldsymbol{P}_D[i-1]\boldsymbol{Q}_{\bar{\boldsymbol{w}}}[i]\\ &\quad + d^*[i]\boldsymbol{P}[i]\boldsymbol{r}[i]\bar{\boldsymbol{w}}^H[i]\boldsymbol{Q}_{\bar{\boldsymbol{w}}}[i]\\ &= \boldsymbol{T}_D[i-1] - \boldsymbol{k}[i]\boldsymbol{P}[i-1]\boldsymbol{P}_D[i-1]\boldsymbol{Q}_{\bar{\boldsymbol{w}}}[i]\\ &\quad + d^*[i]\boldsymbol{k}[i]\bar{\boldsymbol{w}}^H[i]\boldsymbol{Q}_{\bar{\boldsymbol{w}}}[i].\end{aligned} \quad (19)$$

By defining the vector $\boldsymbol{t}[i] = \boldsymbol{Q}_{\bar{\boldsymbol{w}}}[i]\bar{\boldsymbol{w}}[i]$ and using the fact that $\bar{\boldsymbol{r}}^H[i-1] = \boldsymbol{r}^H[i-1]\boldsymbol{T}_D[i-1]$, we arrive at

$$\boldsymbol{T}_D[i] = \boldsymbol{T}_D[i-1] + \boldsymbol{k}[i]\left(d^*[i]\boldsymbol{t}^H[i] - \bar{\boldsymbol{r}}^H[i]\right) \quad (20)$$

where the Kalman gain vector for the computation of $\boldsymbol{T}_D[i]$ is

$$\boldsymbol{k}[i] = \frac{\lambda^{-1}\boldsymbol{P}[i-1]\boldsymbol{r}[i]}{1 + \lambda^{-1}\boldsymbol{r}^H[i]\boldsymbol{P}[i-1]\boldsymbol{r}[i]} \quad (21)$$

and the update for the matrix $\boldsymbol{P}[i]$ employs the matrix inversion lemma [14]

$$\boldsymbol{P}[i] = \lambda^{-1}\boldsymbol{P}[i-1] - \lambda^{-1}\boldsymbol{k}[i]\boldsymbol{r}^H[i]\boldsymbol{P}[i-1]. \quad (22)$$

The vector $\boldsymbol{t}[i]$ is updated as follows:

$$\boldsymbol{t}[i] = \frac{\lambda^{-1}\boldsymbol{Q}_{\bar{\boldsymbol{w}}}[i-1]\bar{\boldsymbol{w}}[i-1]}{1 + \lambda^{-1}\bar{\boldsymbol{w}}^H[i-1]\boldsymbol{Q}_{\bar{\boldsymbol{w}}}[i-1]\bar{\boldsymbol{w}}[i-1]} \quad (23)$$

and the matrix inversion lemma is used to update $\boldsymbol{Q}_{\bar{\boldsymbol{w}}}[i]$ as described by

$$\boldsymbol{Q}_{\bar{\boldsymbol{w}}}[i] = \lambda^{-1}\boldsymbol{Q}_{\bar{\boldsymbol{w}}}[i-1] - \lambda^{-1}\boldsymbol{t}[i]\bar{\boldsymbol{w}}^H[i-1]\boldsymbol{Q}_{\bar{\boldsymbol{w}}}[i-1]. \quad (24)$$

Equations (20)–(24) constitute the first part of the proposed RLS algorithm and are responsible for calculating the projection matrix $\boldsymbol{T}_D[i]$.

To derive a recursive update equation for the reduced-rank filter $\bar{\boldsymbol{w}}[i]$, we consider the expression in (16) with its associated quantities, i.e., the matrix $\bar{\boldsymbol{R}}[i] = \sum_{l=1}^{i} \lambda^{i-l} \bar{\boldsymbol{r}}[l]\bar{\boldsymbol{r}}^H[l]$ and the vector $\bar{\boldsymbol{p}}[i] = \sum_{l=1}^{i} \lambda^{i-l} d^*[l]\bar{\boldsymbol{r}}[l]$. Let us define

$$\bar{\boldsymbol{\Phi}}[i] = \boldsymbol{R}^{-1}[i]$$
$$\bar{\boldsymbol{p}}[i] = \lambda\bar{\boldsymbol{p}}[i-1] + d^*[i]\bar{\boldsymbol{r}}[i] \tag{25}$$

and then, we can rewrite (16) in the following alternative form:

$$\bar{\boldsymbol{w}}[i-1] + \bar{\boldsymbol{k}}[i]\left[d^*[i] - \bar{\boldsymbol{r}}^H[i]\bar{\boldsymbol{w}}[i-1]\right]. \tag{26}$$

By defining $\xi[i] = d[i] - \bar{\boldsymbol{w}}^H[i-1]\bar{\boldsymbol{r}}^H[i]$, we arrive at the proposed RLS algorithm for obtaining $\bar{\boldsymbol{w}}[i]$, i.e.,

$$\bar{\boldsymbol{w}}[i] = \bar{\boldsymbol{w}}[i-1] + \bar{\boldsymbol{k}}[i]\xi^*[i] \tag{27}$$

where the so-called Kalman gain vector is given by

$$\bar{\boldsymbol{k}}[i] = \frac{\lambda^{-1}\bar{\boldsymbol{\Phi}}[i-1]\bar{\boldsymbol{r}}[i]}{1 + \lambda^{-1}\bar{\boldsymbol{r}}^H[i]\bar{\boldsymbol{\Phi}}[i-1]\bar{\boldsymbol{r}}[i]} \tag{28}$$

and the update for the matrix inverse $\bar{\boldsymbol{\Phi}}[i]$ employs the matrix-inversion lemma [14]

$$\bar{\boldsymbol{\Phi}}[i] = \lambda^{-1}\bar{\boldsymbol{\Phi}}[i-1] - \lambda^{-1}\bar{\boldsymbol{k}}[i]\bar{\boldsymbol{r}}^H[i]\bar{\boldsymbol{\Phi}}[i-1]. \tag{29}$$

It should be noted that the proposed RLS algorithm given in (27)–(29) is similar to the conventional RLS algorithm [14], except that it works in a reduced-rank model with a $D \times 1$ input $\bar{\boldsymbol{r}}[i] = \boldsymbol{T}_D^H[i]\boldsymbol{r}[i]$, where the $JM \times D$ matrix $\boldsymbol{T}_D$ is the projection matrix that is responsible for dimensionality reduction.

B. Rank-Selection Algorithm

The performance of the RLS algorithm described in Section V-A depends on the rank D. This motivates the development of methods to automatically adjust D on the basis of the cost function. Unlike prior methods for rank selection that utilize MSWF-based algorithms [25] or the cross-validation (CV) approach used with AVF-based recursions [32], we focus on an approach that determines D based on the LS criterion computed by the filters $\boldsymbol{T}_D[i]$ and $\bar{\boldsymbol{w}}^{(D)}[i]$, where the superscript $^{(D)}$ denotes the rank that is used for the adaptation. Although there are similarities between the algorithm described here and the one reported in [25], the algorithm presented here differs from [25] in that it clearly details the strategy for updating $\boldsymbol{T}_D[i]$ and $\bar{\boldsymbol{w}}^{(D)}[i]$, defines the maximum ($D_{\max}$) and minimum ($D_{\min}$) values for the rank D allowed, and works with extended filters for reduced complexity. The method for automatically selecting the rank of the algorithm is based on the exponentially weighted *a posteriori* LS-type cost function described by

$$\mathcal{C}_{\rm ap}\Big(\boldsymbol{T}_D[i], \bar{\boldsymbol{w}}^{(D)}[i]\Big) = \sum_{l=1}^{i} \alpha^{i-l}\left|d[l] - \bar{\boldsymbol{w}}^{H,(D)}[i]\boldsymbol{T}_D[i]\boldsymbol{r}[l]\right|^2 \tag{30}$$

where α is the forgetting factor, and $\bar{\mathbf{w}}^{(D)}[i]$ is the reduced-rank filter with rank D. For each time interval i, we can select D that minimizes $\mathcal{C}(\boldsymbol{T}_D[i], \bar{\boldsymbol{w}}^{(D)}[i])$, and the exponential weighting factor α is required as the optimal rank varies as a function of the data record. The dimensions of $\boldsymbol{T}_D[i]$ and $\bar{\boldsymbol{w}}^{(D)}[i]$ are extended to $M \times D_{\max}$ and $D_{\max}$, respectively, and the associated matrices $\hat{\bar{\boldsymbol{R}}}[i]$, $\boldsymbol{P}_D[i]$, and $\boldsymbol{Q}_{\bar{\boldsymbol{w}}}[i]$ should be compatible for adaptation. Our strategy is to consider the adaptation with the maximum allowed rank $D_{\max}$ and then perform a search with the aim of finding the best rank within the range $D_{\min}$ to $D_{\max}$. To this end, we define $\boldsymbol{T}_D[i]$ and $\bar{\boldsymbol{w}}^{(D)}[i]$ as follows:

$$\begin{aligned}\boldsymbol{T}_D[i] &= [\,\boldsymbol{t}_1[i] \quad \ldots \quad \boldsymbol{t}_{\mathrm{I}_{\min}}[i] \quad \ldots \quad \boldsymbol{t}_{\mathrm{I}_{\max}}[i]\,]^T\\ \bar{\boldsymbol{w}}^{(D)}[i] &= [\,\bar{w}_1[i] \quad \ldots \quad \bar{w}_{D_{\min}}[i] \quad \ldots \quad \bar{w}_{D_{\max}}[i]\,]^T.\end{aligned} \tag{31}$$

The proposed rank-selection algorithm is given by

$$D_{\rm opt}[i] = \arg \min_{D_{\min} \le d \le D_{\max}} \mathcal{C}_{\rm ap}\left(\boldsymbol{T}_D[i], \bar{\boldsymbol{w}}^{(D)}[i]\right) \tag{32}$$

where d is an integer, and $D_{\min}$ and $D_{\max}$ are the minimum and maximum ranks that are allowed for the reduced-rank filter, respectively. Note that a smaller rank may provide faster adaptation during the initial stages of the estimation procedure, and a greater rank usually yields better steady-state performance. Our studies reveal that the range for which the rank D of the proposed algorithms has a positive impact on the performance of the algorithms is limited, being from $D_{\min} = 3$ to $D_{\max} = 8$ for the reduced-rank filter recursions. These values are rather insensitive to the system load (the number of users) and to the number of array elements and work very well for all scenarios and algorithms examined. The computational complexity of the proposed rank-selection algorithm with extended filters is equivalent to the computation of the cost function in (30) and requires $3(D_{\max} - D_{\min}) + 1$ additions and a sorting algorithm to find the best rank according to (32). An alternative strategy to using extended filters is the deployment of multiple filters with the rank-selection algorithm in (32) that determines the best set of filters for each time interval. Specifically, this approach employs $D_{\max} - D_{\min} + 1$ pairs of filters and has a very high complexity.

A second approach that can be used is a mechanism based on the observation of the columns of $\boldsymbol{T}_D[i]$ and a stopping rule, as reported in [25]. The method performs the following optimization:

$$D_{\rm opt}[i] = \arg\max_{d} \frac{\|P_{\boldsymbol{T}_d}(\boldsymbol{t}_d[i])\|}{\|\boldsymbol{t}_d[i]\|} > \delta \tag{33}$$

where $P_{\boldsymbol{T}_d}(\boldsymbol{x})$ is the orthogonal projection of the vector $\boldsymbol{x}$ onto the subspace $\boldsymbol{T}_d$, and δ is a small positive constant. In [25], the use of a range of values for allowing the selection has not been discussed; however, we found that it is beneficial in terms of complexity to restrict the optimization to an appropriate range of values $D_{\max}$ to $D_{\min}$, as with the previous method.

Another possibility for rank selection is the use of the CV method reported in [32]. This approach selects the filters' lengths that minimize a cost function that is estimated based

TABLE I
COMPUTATIONAL COMPLEXITY OF RLS ALGORITHMS

Algorithm	Additions	Multiplications
Full-rank [14]	$3(JM)^2 - 2JM + 3$	$6(JM)^2 + 2JM + 2$
Proposed	$3(JM)^2 - 2JM + 3$ $6D^2 - 8D + 3$	$7(JM)^2 + 2JM$ $7D^2 + 9D$
MSWF [25]	$D(JM)^2 + (JM)^2 + 6D^2$ $-8D + 2$	$D(JM)^2 + (JM)^2$ $2DJM + 3D + 2$
AVF [32]	$D((JM)^2 + 3(JM-1)^2)$ $+D(5(JM-1)+1)$ $2JM - 1$	$D(4(JM)^2 + 4JM + 1)$ $4JM + 2$

TABLE II
COMPUTATIONAL COMPLEXITY OF THE PROPOSED RANK-SELECTION ALGORITHM WITH MULTIPLE FILTERS

Proposed with Multiple Filters	$2(D_{\max} - D_{\min}) + 1$ $(D_{\max} - D_{\min} + 1)\times$ $(3(JM)^2 - 2JM + 3$ $+6D_{\max}^2 - 8D_{\max} + 3)$	$(D_{\max} - D_{\min} + 1)\times$ $(7(JM)^2 + 2JM$ $+7D_{\max}^2 + 9D_{\max})$

TABLE III
COMPUTATIONAL COMPLEXITY OF THE REMAINING RANK-SELECTION ALGORITHMS

Algorithm	Additions	Multiplications
Proposed with Extended Filter	$2(D_{\max} - D_{\min}) + 1$	$-$
Projection with Stopping Rule [25]	$2(2JM-1)\times$ $(D_{\max} - D_{\min}) + 1$	$((JM)^2 + JM + 1)\times$ $(D_{\max} - D_{\min} + 1)$
CV [32]	$(2JM-1)\times$ $(2(D_{\max} - D_{\min}) + 1)$	$(D_{\max} - D_{\min} + 1)\times$ $JM + 1$

on observations (training data) that have not been used in the process of building the filters themselves as described by

$$\mathcal{C}_{\text{CV}}\left(\boldsymbol{T}_D[i], \bar{\boldsymbol{w}}^{(D)}[i]\right) = \sum_{l=1}^{i} \alpha^{i-l} \left| d(l) - \bar{\boldsymbol{w}}_{(i/l)}^{H,(D)}[i]\boldsymbol{T}_{D,(i/l)}[i]\boldsymbol{r}[l]\right|^2. \quad (34)$$

We consider here the same "leave-one-out" approach as in [32]. For a given data record of size i, the CV approach chooses the filter $\bar{\boldsymbol{w}}^{H,(D)}[i]$ that performs the following optimization:

$$D_{\text{opt}}[i] = \arg \min_{d\in\{1,2,\ldots\}} \mathcal{C}_{\text{CV}}\left(\boldsymbol{T}_d[i], \bar{\boldsymbol{w}}^{(d)}[i]\right). \quad (35)$$

The main difference between this and the other algorithms presented lies in the use of CV, which leaves one sample out in the process, and the use of the constraint on the allowed filter lengths. In the simulations, we will compare the rank-selection algorithms and discuss their advantages and disadvantages.

C. Computational Complexity

Here, we detail the computational complexity requirements of the proposed RLS algorithms and compare them with those of existing algorithms. We also provide the computational complexity of the proposed and existing rank-selection algorithms. The computational complexity expressed in terms of additions and multiplications is depicted in Table I for the RLS algorithms, the complexity of the proposed rank-selection algorithm with multiple filters, including the proposed RLS algorithm, is illustrated in Table II, and that of the remaining rank-selection techniques is given in Table III.

In the case of the proposed reduced-rank RLS algorithm, the complexity is quadratic with $(JM)^2$ and D^2. This corresponds to a complexity that is slightly higher than the one observed for the full-rank RLS algorithm, provided D is significantly smaller

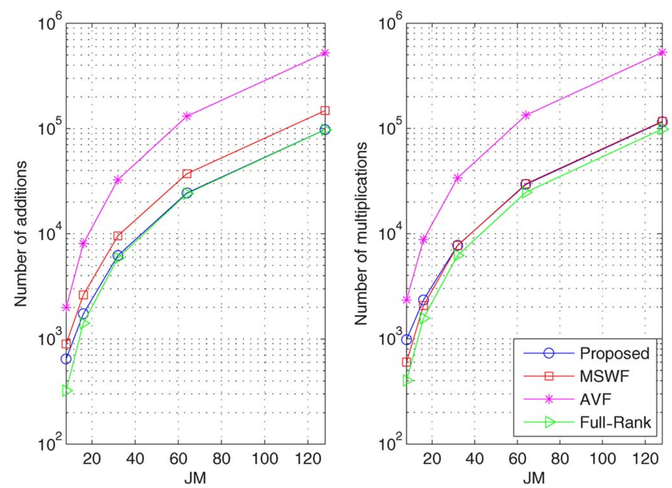

Fig. 2. Complexity in terms of additions and multiplications against the number of input samples JM and $D = 4$.

than JM, and comparable with the cost of the MSWF-RLS [25] and the AVF [32]. To illustrate the main trends in what concerns the complexity of the proposed and analyzed algorithms, we show in Fig. 2 the complexity against the number of input samples JM. The curves indicate that the proposed reduced-rank RLS algorithm has a complexity that is lower than the MSWF-RLS algorithm [25] and the AVF [32], whereas it remains at the same level of the full-rank RLS algorithm.

The proposed rank-selection algorithm with multiple filters has a number of arithmetic operations that are substantially higher than the other compared methods since it requires the computation of $D_{\max} - D_{\min} + 1$ pairs of filters with the proposed RLS algorithms simultaneously. We separately show the overall cost of this algorithm in Table II. The computational complexity of the remaining rank-selection algorithms, including the proposed and existing rank-selection algorithms, is shown in Table III. From Table III, we can notice that the proposed rank-selection algorithm with extended filters is significantly less complex than the existing methods based on projection with stopping rule [25] and the CV approach [32]. Specifically, the proposed rank-selection algorithm with extended filters only requires $2(D_{\max} - D_{\min})$ additions, as depicted in the first row of Table III. To this cost, we must add the operations that are required by the proposed RLS algorithm, whose complexity is shown in the second row of Table I using $D_{\max}$ according to the procedure outlined in Section V-B. The complexities of the MSWF and AVF algorithms are detailed in the third and fourth rows of Table I. For their operation with rank-selection algorithms, a designer must add their complexities in Table I to the complexity of the rank-selection algorithm of interest, as shown in Table III.

VI. Analysis

Here, we conduct a convexity analysis of the proposed optimization that is responsible for designing the filters $\boldsymbol{T}_D[i]$ and $\bar{\boldsymbol{w}}[i]$ of the proposed scheme. We show that the proposed optimization leads to a problem with multiple solutions, and we discuss the properties of the method. In particular, we conjecture that it leads to a problem with multiple and possibly identical minimum points. This is corroborated by numerous studies that verify that the method is insensitive to different initializations (except for the case when $\boldsymbol{T}_D[i]$ is a null matrix and annihilates the received signal) and that it always converges to the same point of minimum. We also establish the convergence of the proposed optimization algorithm, showing that the sequence of filters $\boldsymbol{T}_D[i]$ and $\bar{\boldsymbol{w}}[i]$ produces a sequence of outputs that is bounded.

A. Convexity Analysis of the Proposed Method

Here, we carry out a convexity analysis of the proposed reduced-rank scheme and LS optimization algorithm. Our approach is based on expressing the output of the proposed scheme in a convenient form that renders itself to analysis. Let us consider the proposed optimization method in (13) and express it by an expanded cost function

$$\begin{aligned}\mathcal{C}\left(\boldsymbol{T}_D[i], \bar{\boldsymbol{w}}^H[i]\right) &= \sum_{l=1}^{i} \lambda^{i-l} \left|d[l] - \bar{\boldsymbol{w}}^H[i]\boldsymbol{T}_D^H[i]\boldsymbol{r}[l]\right|^2\Big] \\ &= \sum_{l=1}^{i} \lambda^{i-l} \left|d[l]\right|^2 - \sum_{l=1}^{i} \lambda^{i-l}\bar{\boldsymbol{w}}^H[i]\boldsymbol{T}_D^H[i]d^*[l]\boldsymbol{r}[l] \\ &\quad - \sum_{l=1}^{i} \lambda^{i-l} d[l]\boldsymbol{r}^H[l]\boldsymbol{T}_D[i]\bar{\boldsymbol{w}}[i] \\ &\quad + \sum_{l=1}^{i} \lambda^{i-l}\bar{\boldsymbol{w}}^H[i]\boldsymbol{T}_D^H[i]\boldsymbol{r}[l]\boldsymbol{r}^H[l]\boldsymbol{T}_D[i]\bar{\boldsymbol{w}}[i]. \end{aligned} \tag{36}$$

To proceed, let us express $x[i]$ in an alternative and more convenient form as

$$\begin{aligned} x[i] &= \bar{\boldsymbol{w}}^H[i]\boldsymbol{T}_D^H[i]\boldsymbol{r}[i] = \bar{\boldsymbol{w}}^H[i]\sum_{d=1}^{D}\boldsymbol{T}_D^H[i]\boldsymbol{r}[i]\boldsymbol{v}_d \\ &= \bar{\boldsymbol{w}}^H[i]\begin{bmatrix} \boldsymbol{r}[i] & 0 & \dots & 0 \\ 0 & \boldsymbol{r}[i] & \ddots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & \dots & \dots & \boldsymbol{r}[i]\end{bmatrix}^T \begin{bmatrix} \boldsymbol{s}_1^*[i] \\ \boldsymbol{s}_2^*[i] \\ \vdots \\ \boldsymbol{s}_D^*[i]\end{bmatrix} \\ &= \bar{\boldsymbol{w}}^H[i]\boldsymbol{\Re}^T[i]\boldsymbol{s}_v^*[i] \end{aligned} \tag{37}$$

where $\boldsymbol{\Re}[i]$ is a $DJM \times D$ block diagonal matrix with the input data vector $\boldsymbol{r}[i]$, $\boldsymbol{s}_v^*[i]$ is a $DJM \times 1$ vector with the columns of $\boldsymbol{T}_D[i]$ stacked on top of each other, and the $D \times 1$ vector $\boldsymbol{v}_d$ contains a 1 in the dth position and zeros elsewhere.

To analyze the proposed joint-optimization procedure, we can rearrange the terms in $x[i]$ and define a single $D(JM+1) \times 1$ parameter vector $\boldsymbol{q}[i] = [\bar{\boldsymbol{w}}^T[i]\ \boldsymbol{s}_v^T[i]]^T$. We can, therefore, further express $x[i]$ as

$$\begin{aligned} x[i] &= \boldsymbol{q}^H[i]\begin{bmatrix} \boldsymbol{0}_{D\times 1} & \boldsymbol{0}_{D\times DJM} \\ \boldsymbol{\Re}[i] & \boldsymbol{0}_{DJM\times DJM}\end{bmatrix}\boldsymbol{q}[i] \\ &= \boldsymbol{q}^H[i]\boldsymbol{U}[i]\boldsymbol{q}[i] \end{aligned} \tag{38}$$

where $\boldsymbol{U}[i]$ is a $D(JM+1) \times D(JM+1)$ matrix that contains $\boldsymbol{\Re}[i]$. At this stage, we can alternatively express the cost function in (36) as

$$\mathcal{C}\left(\boldsymbol{q}[i]\right) = \sum_{l=1}^{i}\left|d[l] - \boldsymbol{q}^H[i]\boldsymbol{U}[l]\boldsymbol{q}[i]\right|^2. \tag{39}$$

We can examine the convexity of the above by computing the Hessian $(\boldsymbol{H})$ with respect to $\boldsymbol{q}[i]$ using the expression [42]

$$\boldsymbol{H} = \frac{\partial}{\partial \boldsymbol{q}^H[i]}\frac{\partial(\mathcal{C}\left(\boldsymbol{q}[i]\right))}{\partial \boldsymbol{q}[i]} \tag{40}$$

and testing if the terms are positive semidefinite. Specifically, $\boldsymbol{H}$ is positive semidefinite if $\boldsymbol{a}^H\boldsymbol{H}\boldsymbol{a} \geq 0$ for all nonzero $\boldsymbol{a} \in \boldsymbol{C}^{D(JM+1)\times D(JM+1)}$ [41], [42]. Therefore, the optimization problem is convex if the Hessian $\boldsymbol{H}$ is positive semidefinite.

Evaluating the partial differentiation in the expression given in (40) yields

$$\begin{aligned} \boldsymbol{H} &= \sum_{l=1}^{i}\left(\boldsymbol{q}^H[i]\boldsymbol{U}[l]\boldsymbol{q}[i] - d^*[l]\right)\boldsymbol{U}[l] \\ &\quad + \sum_{l=1}^{i}\boldsymbol{U}^H[l]\boldsymbol{q}[i]\boldsymbol{q}[i]^H\boldsymbol{U}[l] \\ &\quad + \sum_{l=1}^{i}\left(\boldsymbol{q}^H[i]\boldsymbol{U}[l]\boldsymbol{q}[i] - d[l]\right)\boldsymbol{U}^H[l] \\ &\quad + \sum_{l=1}^{i}\boldsymbol{U}[l]\boldsymbol{q}[i]\boldsymbol{q}[i]^H\boldsymbol{U}^H[l]. \end{aligned} \tag{41}$$

By examining $\boldsymbol{H}$, we verify that the second and fourth terms are positive semidefinite, whereas the first and third terms are indefinite. Therefore, the optimization problem cannot be classified as convex. It is, however, important to remark that our studies indicate that there are no local minima, and there exist multiple solutions (that are conjectured to be identical).

To support this claim, we have checked the impact on the proposed algorithms of different initializations. This study confirmed that the algorithms are not subject to performance degradation due to the initialization, although we have to bear in mind that the initialization $\boldsymbol{T}_D(0) = \boldsymbol{0}_{JM\times D}$ annihilates the signal and must be avoided. We have also studied a particular case of the proposed scheme when $JM = 1$ and $D = 1$, which yields the cost function

$$\mathcal{C}(\boldsymbol{T}_D, \bar{\boldsymbol{w}}) = E\left[\left|d - \bar{w}T_D r\right|^2\right]\Big]. \tag{42}$$

By choosing T_D (the "scalar" projection) fixed with D equal to 1, it is evident that the resulting function $\mathcal{C}(\bar{w}, T_D = 1, r) = |d - w^* r|^2]$ is a convex one. In contrast to that, for a time-varying projection T_D, the plots of the function indicate that the function is no longer convex, but it also does not exhibit local minima. The problem at hand can be generalized to the vector case; however, we can no longer verify the existence of local minima due to the multidimensional surface. This remains as an interesting open problem to be studied.

B. Proof of Convergence of the Method

Here, we show that the proposed reduced-rank algorithm converges globally and exponentially to the optimal reduced-rank estimator [15], [22], [23]. An issue that remains an open problem to be investigated is the transient behavior of the proposed method, which corresponds to the fact that the most significant difference between the proposed and existing (MSWF and AVF) methods is on the transient performance. To our knowledge, there exists no result for the transient analysis of the MSWF and AVF methods, although it has been reported (and verified in our studies) that the AVF [32] has superior convergence performance over the MSWF.

As discussed in Section VI-A, the optimal solutions $\boldsymbol{T}_{D,\mathrm{opt}}$ and $\bar{\boldsymbol{w}}_{\mathrm{opt}}$ are not unique. However, the desired product of the optimal solutions, i.e., $\boldsymbol{w}_{\mathrm{opt}} = \boldsymbol{T}_{D,\mathrm{opt}}\bar{\boldsymbol{w}}_{\mathrm{opt}}$, is known and given by $\boldsymbol{R}^{-1/2}(\boldsymbol{R}^{-1/2}\boldsymbol{p})_{1:D}$ [14], [22], [23], where $\boldsymbol{R}^{-1/2}$ is the square root of the input data covariance matrix, and the subscript $1:D$ denotes truncation of the subspace.

To proceed with our proof, let us rewrite the expressions in (15) and (16) for time instant 0 as follows:

$$\boldsymbol{R}[0]\boldsymbol{T}_D[0]\boldsymbol{R}_w[0] = \boldsymbol{P}_D[0] = \boldsymbol{p}[0]\bar{\boldsymbol{w}}^H[0] \tag{43}$$

$$\bar{\boldsymbol{R}}[0]\bar{\boldsymbol{w}}[1] = \boldsymbol{T}_D^H[0]\boldsymbol{R}[0]\boldsymbol{T}_D[0]\bar{\boldsymbol{w}}[1] = \bar{\boldsymbol{p}}[0]. \tag{44}$$

Using (43), we can obtain the following relation:

$$\boldsymbol{R}_w[0] = \left(\boldsymbol{T}_D^H[0]\boldsymbol{R}^2[0]\boldsymbol{T}_D[0]\right)^{-1}\boldsymbol{T}_D^H[0]\boldsymbol{R}[0]\boldsymbol{p}[0]\bar{\boldsymbol{w}}^H[0]. \tag{45}$$

Substituting the above result for $\boldsymbol{R}_w[0]$ into the expression in (43), we get a recursive expression for $\boldsymbol{T}_D[0]$, i.e.,

$$\boldsymbol{T}_D[0] = \boldsymbol{R}[0]^{-1}\boldsymbol{p}[0]\bar{\boldsymbol{w}}^H[0]\left(\boldsymbol{T}_D^H[0]\boldsymbol{R}[0]\boldsymbol{p}[0]\bar{\boldsymbol{w}}^H[0]\right)^{-1} \times \left(\boldsymbol{T}_D^H[0]\boldsymbol{R}^2[0]\boldsymbol{T}_D[0]\right)^{-1}. \tag{46}$$

Using (44), we can express $\bar{\boldsymbol{w}}[1]$ as

$$\bar{\boldsymbol{w}}[1] = \left(\boldsymbol{T}_D^H[0]\boldsymbol{R}[0]\boldsymbol{T}_D[0]\right)^{-1}\boldsymbol{T}_D^H[0]\boldsymbol{p}[0]. \tag{47}$$

Taking into account the relation $\boldsymbol{w}[1] = \boldsymbol{T}_D[1]\bar{\boldsymbol{w}}[1]$, we obtain

$$\boldsymbol{w}[1] = \boldsymbol{R}[1]^{-1}\boldsymbol{p}[1]\bar{\boldsymbol{w}}^H[1]\left(\boldsymbol{T}_D^H[1]\boldsymbol{R}[1]\boldsymbol{p}[1]\bar{\boldsymbol{w}}^H[1]\right)^{-1} \cdot \left(\boldsymbol{T}_D^H[1]\boldsymbol{R}^2[1]\boldsymbol{T}_D[1]\right)^{-1}\left(\boldsymbol{T}_D^H[0]\boldsymbol{R}[0]\boldsymbol{T}_D[0]\right)^{-1}\boldsymbol{T}_D^H[0]\boldsymbol{p}[0]. \tag{48}$$

More generally, we can express the proposed reduced-rank LS algorithm by the following recursion:

$$\begin{aligned}\boldsymbol{w}[i] &= \boldsymbol{T}_D[i]\bar{\boldsymbol{w}}[i]\\ &= \boldsymbol{R}[i]^{-1}\boldsymbol{p}[i]\bar{\boldsymbol{w}}^H[i]\left(\boldsymbol{T}_D^H[i]\boldsymbol{R}[i]\boldsymbol{p}[i]\bar{\boldsymbol{w}}^H[i]\right)^{-1}\\ &\quad\cdot\left(\boldsymbol{T}_D^H[i]\boldsymbol{R}^2[i]\boldsymbol{T}_D[i]\right)^{-1}\\ &\quad\cdot\left(\boldsymbol{T}_D^H[i-1]\boldsymbol{R}[i-1]\boldsymbol{T}_D[i-1]\right)^{-1}\boldsymbol{T}_D^H[i-1]\boldsymbol{p}[i-1].\end{aligned} \tag{49}$$

Since the optimal reduced-rank filter can be described by the SVD of $\boldsymbol{R}^{-1/2}\boldsymbol{p}$ [15], [22], [23], where $\boldsymbol{R}^{-1/2}$ is the square root of the covariance matrix $\boldsymbol{R}$, and $\boldsymbol{p}$ is the cross-correlation vector, we then have

$$\boldsymbol{R}^{-1/2}\boldsymbol{p} = \boldsymbol{\Phi}\boldsymbol{\Lambda}\boldsymbol{\Phi}^H\boldsymbol{p}. \tag{50}$$

Considering that there exists some $\boldsymbol{w}[0]$ such that the randomly selected $\boldsymbol{T}_D[0]$ can be written as [22], [23]

$$\boldsymbol{T}_D[0] = \boldsymbol{R}^{-1/2}\boldsymbol{\Phi}\boldsymbol{w}[0]. \tag{51}$$

Substituting (51) and using (50) in (49) and manipulating the algebraic expressions, we can express (49) in a more compact way that is suitable for analysis, as given by

$$\boldsymbol{w}[i] = \boldsymbol{\Lambda}^2\boldsymbol{w}[i-1]\left(\boldsymbol{w}^H[i-v1]\boldsymbol{\Lambda}^2\boldsymbol{w}[i-1]\right)^{-1}\boldsymbol{w}^H[i-1]\boldsymbol{w}[i-1]. \tag{52}$$

The above expression can be decomposed as follows:

$$\boldsymbol{w}[i] = \boldsymbol{Q}[i]\,\boldsymbol{Q}[i-1]\,\ldots\,\boldsymbol{Q}[1]\,\boldsymbol{w}[0] \tag{53}$$

where

$$\boldsymbol{Q}[i] = \boldsymbol{\Lambda}^{2i}\boldsymbol{w}[0]\left(\boldsymbol{w}^H[0]\boldsymbol{\Lambda}^{4i-2}\boldsymbol{w}[0]\right)^{-1}\boldsymbol{w}^H[0]\boldsymbol{\Lambda}^{2i-2}. \tag{54}$$

At this point, we need to establish that the norm of $\boldsymbol{T}_D[i]$ for all i is both lower and upper bounded, i.e., $0 < \|\boldsymbol{T}_D[i]\| < \infty$ for all i and that $\boldsymbol{w}[i] = \boldsymbol{T}_D[i]\bar{\boldsymbol{w}}[i]$ exponentially approaches $\boldsymbol{w}_{\mathrm{opt}}[i]$ as i increases. Due to the linear mapping, the boundedness of $\boldsymbol{T}_D[i]$ is equivalent to that of $\boldsymbol{w}[i]$. Therefore, we have, upon convergence, $\boldsymbol{w}^H[i]\boldsymbol{w}[i-1] = \boldsymbol{w}^H[i-1]\boldsymbol{w}[i-1]$. Since $\|\boldsymbol{w}^H[i]\boldsymbol{w}[i-1]\| \le \|\boldsymbol{w}[i-1]\|\|\boldsymbol{w}[i]\|$ and $\|\boldsymbol{w}^H[i-1]\boldsymbol{w}[i-1]\| = \|\boldsymbol{w}[i-1]\|^2$, the relation $\boldsymbol{w}^H[i]\boldsymbol{w}[i-1] = \boldsymbol{w}^H[i-1]\boldsymbol{w}[i-1]$ implies $\|\boldsymbol{w}[i]\| > \|\boldsymbol{w}[i-1]\|$, and hence

$$\|\boldsymbol{w}[\infty]\| \ge \|\boldsymbol{w}[i]\| \ge \|\boldsymbol{w}[0]\|. \tag{55}$$

To show that the upper bound $\|\boldsymbol{w}[\infty]\|$ is finite, let us express the $JM \times JM$ matrix $\boldsymbol{Q}[i]$ as a function of the $JM \times 1$ vector $\boldsymbol{w}[i] = \left[\begin{smallmatrix}\boldsymbol{w}_1[i]\\ \boldsymbol{w}_2[i]\end{smallmatrix}\right]$ and the $JM \times JM$ matrix $\boldsymbol{\Lambda} = \begin{bmatrix}\boldsymbol{\Lambda}_1 & \\ & \boldsymbol{\Lambda}_2\end{bmatrix}$. Substituting the previous expressions of $\boldsymbol{w}[i]$ and $\boldsymbol{\Lambda}$ into $\boldsymbol{Q}[i]$ given in (54), we obtain

$$\boldsymbol{Q}[i] = \begin{bmatrix}\boldsymbol{\Lambda}_1^{2i}\boldsymbol{w}_1[0]\\ \boldsymbol{\Lambda}_2^{2i}\boldsymbol{w}_2[0]\end{bmatrix}\left(\boldsymbol{w}_1^H[0]\boldsymbol{\Lambda}_1^{4i-2}\boldsymbol{w}_1[0] + \boldsymbol{w}_2^H[0]\boldsymbol{\Lambda}_2^{4i-2}\boldsymbol{w}_2[0]\right)^{-1} \times \begin{bmatrix}\boldsymbol{w}_1^H[0]\boldsymbol{\Lambda}_1^{2i-2}\\ \boldsymbol{w}_2^H[0]\boldsymbol{\Lambda}_2^{2i-2}\end{bmatrix}. \tag{56}$$

Applying the matrix identity $(\boldsymbol{A}+\boldsymbol{B})^{-1}=\boldsymbol{A}^{-1}-\boldsymbol{A}^{-1}\boldsymbol{B}(\boldsymbol{I}+\boldsymbol{A}^{-1}\boldsymbol{B})^{-1}\boldsymbol{A}^{-1}$ to the decomposed $\boldsymbol{Q}[i]$ in (56) and making i large, we get

$$\boldsymbol{Q}[i] = \mathrm{diag}(\underbrace{1\ldots 1}_{D}\ \underbrace{0\ldots 0}_{JM-D}) + \mathrm{O}\left(\epsilon[i]\right) \tag{57}$$

where $\epsilon[i] = (\lambda_{r+1}/\lambda_r)^{2i}$, with λ_{r+1} and λ_r the $(r+1)$th and rth largest singular values of $\boldsymbol{R}^{-1/2}\boldsymbol{p}$, respectively. From (57), it follows that for some positive constant k, we have $\|\boldsymbol{w}[i]\| \leq 1 + k\epsilon[i]$. From (53), we obtain

$$\begin{aligned}\|\boldsymbol{w}[\infty]\| &\leq \|\boldsymbol{Q}[\infty]\| \ldots \|\boldsymbol{Q}[2]\|\ \|\boldsymbol{Q}[1]\|\ \|\boldsymbol{Q}[0]\| \\ &\leq \|\boldsymbol{w}[0]\| \prod_{i=1}^{\infty}\left(1+k\epsilon[i]\right) \\ &= \|\boldsymbol{w}[0]\| \exp\left(\sum_{i=1}^{\infty} log\left(1+k\epsilon[i]\right)\right) \\ &\leq \|\boldsymbol{w}[0]\| \exp\left(\sum_{i=1}^{\infty} k\epsilon[i]\right) \\ &= \|\boldsymbol{w}[0]\| \exp\left(\frac{k}{1-(\lambda_{r+1}/\lambda_r)^2}\right).\end{aligned} \tag{58}$$

With the development above, the norm of $\boldsymbol{w}[i]$ is proven to be both lower and upper bounded. Once this is established, the expression in (49) converges for large i to the reduced-rank Wiener filter. This can be verified by equating the terms of (52), which yields

$$\begin{aligned}\boldsymbol{w}[i] &= \boldsymbol{R}[i]^{-1}\boldsymbol{p}[i]\bar{\boldsymbol{w}}^H[i]\left(\boldsymbol{T}_D^H[i]\boldsymbol{R}[i]\boldsymbol{p}[i]\bar{\boldsymbol{w}}^H[i]\right)^{-1} \\ &\quad \times \left(\boldsymbol{T}_D^H[i]\boldsymbol{R}^2[i]\boldsymbol{T}_D[i]\right)^{-1} \\ &\quad \cdot \left(\boldsymbol{T}_D^H[i-1]\boldsymbol{R}[i-1]\boldsymbol{T}_D[i-1]\right)^{-1}\boldsymbol{T}_D^H[i-1]\boldsymbol{p}[i-1] \\ &= \boldsymbol{R}^{-1/2}\boldsymbol{\Phi}_1\boldsymbol{\Lambda}_1\boldsymbol{\Phi}_1^H\boldsymbol{p} + \mathrm{O}\left(\epsilon[i]\right)\end{aligned} \tag{59}$$

where $\boldsymbol{\Phi}_1$ is a $JM \times D$ matrix with the D largest eigenvectors of $\boldsymbol{R}$, and $\boldsymbol{\Lambda}_1$ is a $D \times D$ matrix with the largest eigenvalues of $\boldsymbol{R}$.

VII. Simulations

The performance of the proposed scheme and algorithms is assessed in terms of the uncoded bit error rate (BER) via simulations for space–time interference suppression in a DS-CDMA system. We consider dynamic fading situations and perfect synchronization, and the proposed and existing adaptive algorithms are employed to adjust the filters and track the channel variations. Specifically, in our proposed reduced-rank STAP, the output of the receiver $x[i]$ is the input to a slicer that makes the decision about the transmitted symbol $\hat{b}_k[i]$ for user k as follows:

$$\hat{b}_k[i] = Q\left(x[i]\right) = Q\left(\hat{\mathbf{w}}^H[i]\boldsymbol{T}_D^H[i]\boldsymbol{r}[i]\right) \tag{60}$$

where $Q(\cdot)$ is the function that implements the slicer, and the kth user is assumed to be user 1.

For all simulations, we use the initial values $\bar{\boldsymbol{w}}[0] = [1\ 0\ \ldots\ 0]^T$ and $\boldsymbol{T}_D[0] = [\boldsymbol{I}_D\ \boldsymbol{0}_{D,JM-D}]^T$. We assume $L = 9$ to be an upper bound and employ quaternary phase-shift keying symbols and three-path channels with a power delay profile [43] given by 0, -3, and -6 dB, where, in each run, the spacing between paths is obtained from a discrete uniform random variable between 1 and 2 chips, and the experiments are averaged over 200 runs. The power and the phase of each path are time-varying and follow Clarke's model [43]. This procedure corresponds to the generation of independent sequences of correlated unit power Rayleigh random variables for each path. The DoAs of the interferers and the desired user are uniformly distributed in $(0, 2\pi/3)$. The system has a power distribution among the users for each run that follows a lognormal distribution with an associated standard deviation equal to 1.5 dB. We compare the proposed scheme with the full-rank [14], MSWF [25], and AVF [28] techniques for the design of linear space–time receivers, as well as the rank-selection algorithms reported in [25] and [32] with the proposed rank-selection techniques.

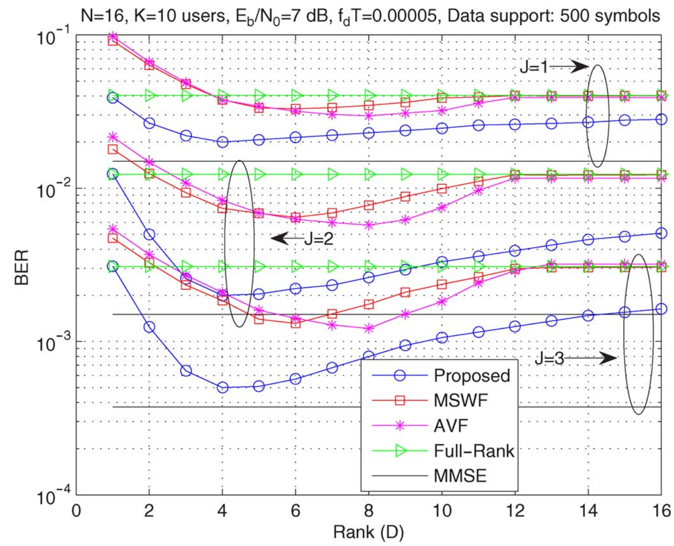

Fig. 3. BER performance versus rank D.

In the first scenario, we consider the BER performance versus the rank D with optimized parameters (forgetting factors $\lambda = 0.998$) for all schemes. The results in Fig. 3 indicate that the best rank for the proposed scheme is $D = 4$ for a data record of 500 symbols, as it is very close to the optimal linear MMSE estimator. Studies with systems with different processing gains and loads show that D does not significantly vary with either the system size or the load. However, it should be remarked that considerable performance gains can be obtained with an automatic rank-adaptation algorithm to fine tune the used rank.

In a second experiment, the BER convergence performance in a mobile communication situation is shown in Fig. 4. The channel coefficients are obtained with Clarke's model [43], and the adaptive estimators of all methods are trained with 200 symbols and are then switched to decision-directed mode. The results show that the proposed scheme has considerably better performance than the existing approaches and is able to adequately track the desired signal. In particular, the proposed reduced-rank algorithm converges in 100 symbols for the case of $J = 1$, in about 200 symbols for the case of $J = 2$, and in about 400 symbols for $J = 3$. This is substantially faster than

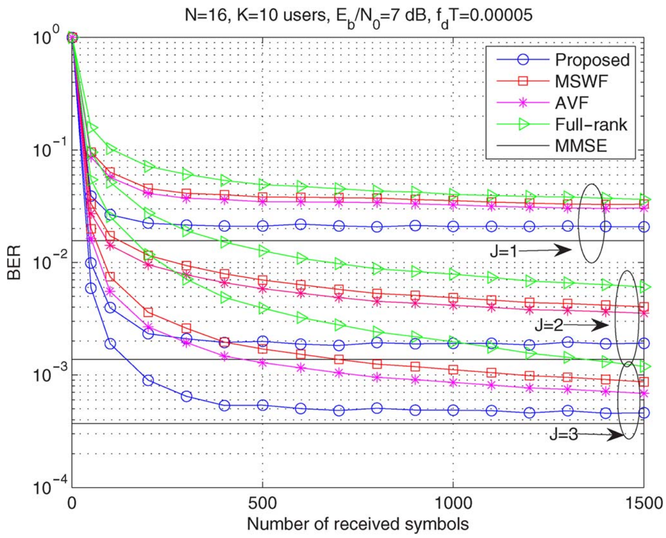

Fig. 4. BER performance versus the number of received symbols.

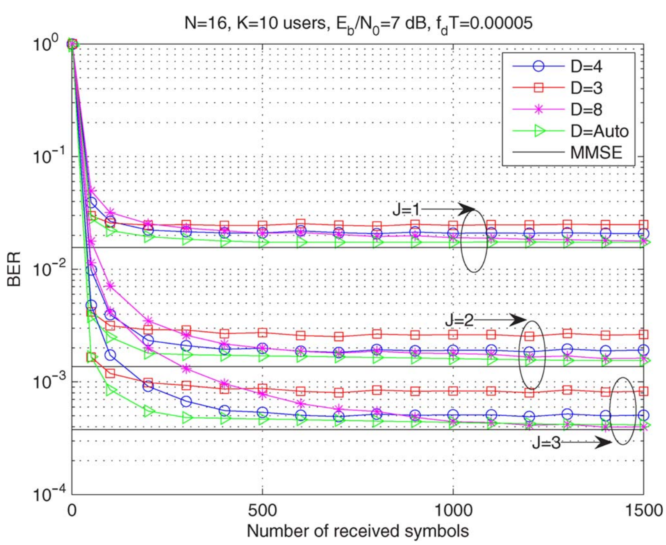

Fig. 5. BER performance versus the number of received symbols with automatic rank adaptation.

the existing reduced-rank schemes, namely, the MSWF and the AVF (which are known to have the best performance available in the area), and the full-rank RLS algorithm.

In practice, the rank D can be adapted to obtain fast convergence and ensure good steady-state performance and tracking after convergence. To this end, we developed the automatic rank-selection algorithm in Section V. We will assess this algorithm in a scenario that is identical to the previous experiment. The results in Fig. 5 show that significant gains can be obtained from the use of the automatic rank-selection algorithm. Specifically, we can notice that the proposed reduced-rank algorithm has very fast convergence with $D = 3$, although it does not provide steady-state performance that is close to the full-rank optimal linear MMSE estimator. When the proposed reduced-rank algorithm employs $D = 8$, the convergence is notably slower, although it is able to approach the full-rank optimal linear MMSE estimator in steady state, as shown in Fig. 5 and evidenced in our studies. Interestingly,

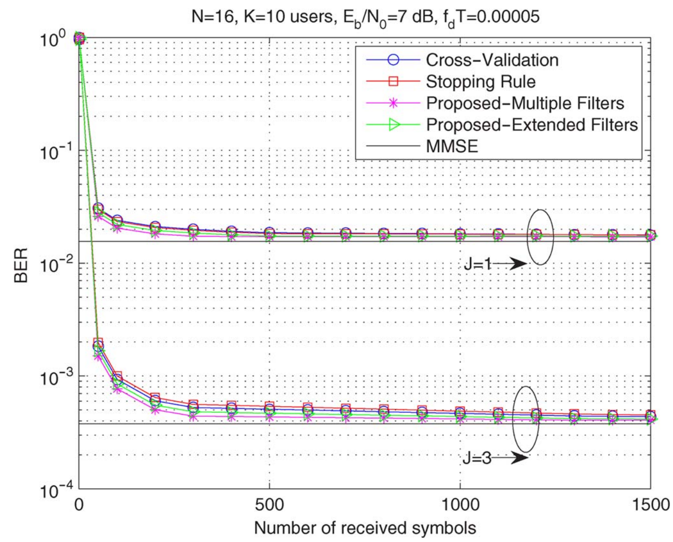

Fig. 6. BER performance versus the number of received symbols with different automatic rank adaptation algorithms and the proposed reduced-rank scheme and algorithm.

when equipped with the proposed automatic rank-selection algorithm, the proposed reduced-rank RLS algorithm achieves convergence performance that is as good as with $D = 3$ and steady-state performance that is equivalent to that with $D = 8$. Another important issue is that the differences in performance are more pronounced for larger filters when the usefulness of the automatic rank-selection algorithm becomes clearer.

To assess the performance of the proposed rank-selection algorithms, we consider the scenario of the previous experiment with $J = 1$ and 3 and compare the rank-selection algorithms based on a stopping criterion [25], the CV method in [32], and the proposed LS-based method with two variations, namely, the multiple filters and the extended filter approaches. The results shown in Fig. 6 indicate that the LS-based methods are slightly better than the other techniques. The CV approach has the advantage that it does not require the setting of $D_{\min}$ and $D_{\max}$; however, it may perform a search over a higher range of values that leads to higher complexity. The remaining techniques operate with $D_{\min} = 3$ and $D_{\max} = 8$. The method with a stopping rule has performance that is slightly worse than the remaining schemes, and its complexity is higher than those of the LS-based techniques due to the computation of the orthogonal projection.

At this point, we will consider a study of the BER performance against the normalized fading rate of the channel f_dT in the experiment shown in Fig. 7. We assess the performance of the receivers with a data record of 1000 symbols of training. The proposed algorithm is equipped with the automatic rank-selection algorithm, and the MSWF and AVF algorithms are also equipped with the rank-adaptation techniques reported in [25] and [32], respectively. We observe from the curves in Fig. 7 that the proposed reduced-rank algorithm obtains substantial gains in BER performance over the existing MSWF and AVF algorithms and the full-rank RLS algorithm. We can notice that as the channel becomes more hostile, the performance of the analyzed algorithms degrades, indicating that the adaptive techniques are encountering difficulties in dealing with the

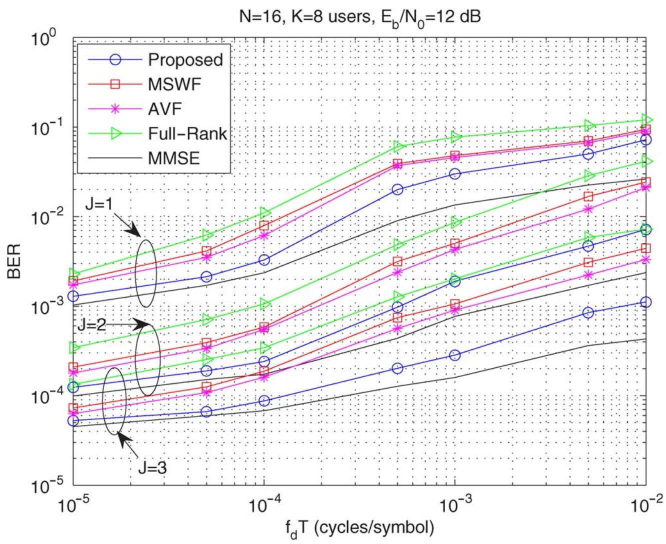

Fig. 7. BER performance versus the number of received symbols.

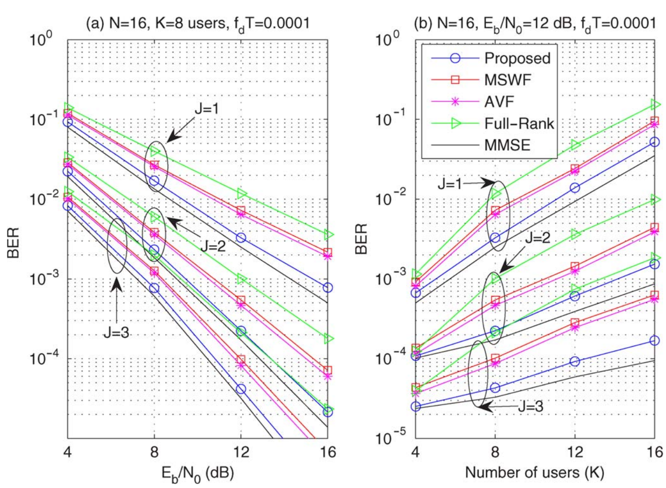

Fig. 8. BER performance against (a) E_b/N_0 (in decibels) and (b) number of users K for different techniques.

changing environment and interference. This behavior is more pronounced when the algorithms have to adjust filters with more coefficients, e.g., for more antenna elements $(J = 2, 3)$. In this regard, the reduced-rank algorithms obtain significant gains over the full-rank RLS algorithm, and in particular, the proposed reduced-rank algorithm achieves the best performance among them.

The last experiment shows the BER performance versus the E_b/N_0 and the number of users (K), which is illustrated in Fig. 8. In this scenario, all algorithms are trained with 200 symbols and are switched to decision-directed mode for processing another 1500 symbols. The curves are obtained after 5000 runs. The proposed algorithm is equipped with the automatic rank-selection algorithm, and the MSWF and AVF techniques are also equipped with the rank-adaptation methods reported in [25] and [32], respectively. The results confirm the excellent performance of the proposed reduced-rank algorithm, which can approach the performance of the optimal MMSE full-rank linear estimator (simply denoted as the MMSE) that assumes the knowledge of the channels, the DoAs, and the noise variance. In particular, the proposed reduced-rank algorithm can save up to 2 dB in E_b/N_0 in comparison with the existing reduced-rank techniques for the same BER performance, whereas it can accommodate up to four more users than the MSWF and the AVF for the same BER performance. Interestingly, the performance of the optimal reduced-rank linear MMSE estimator [15] that assumes the knowledge of $\boldsymbol{R}$ and employs SVD is quite similar to that of the optimal full-rank one. For this reason, we only show the performance of the full-rank optimal linear MMSE estimator.

VIII. Conclusion

We have proposed a reduced-rank scheme based on joint iterative optimization of parameter vectors. In the proposed scheme, the full-rank adaptive filters are responsible for estimating the subspace projection rather than the desired signal, which is estimated by a small reduced-rank filter. We have developed a computationally efficient RLS algorithm for estimating the parameters of the proposed scheme and an automatic rank-selection algorithm to compute the rank of the proposed RLS algorithm. The proposed algorithms require neither an SVD for dimensionality reduction nor any knowledge about the order of the reduced-rank model. The results for space–time interference suppression in a DS-CDMA system show performance that is significantly better than existing schemes and close to the full-rank optimal linear MMSE estimator in dynamic and hostile environments. The proposed algorithms can be employed in a variety of applications, including spread-spectrum and multiple-input–multiple-output systems, wireless networks, cooperative communications, and navigation receivers.

References

[1] A. J. Paulraj and C. B. Papadias, "Space–time processing for wireless communications," *IEEE Signal Process. Mag.*, vol. 14, no. 6, pp. 49–83, Nov. 1997.

[2] J. R. Guerci, *Space–Time Adaptive Processing for Radar*. Norwood, MA: Artech House, 2003.

[3] R. Klemm, *Principles of Space–Time Adaptive Processing*. London, U.K.: IEE, 2002.

[4] P. Xiong, M. Medley, and S. Batalama, "Spatial and temporal processing for global navigation satellite systems: The GPS receiver paradigm," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 39, no. 4, pp. 1471–1484, Oct. 2003.

[5] J. G. Proakis, *Digital Communications,* 4th ed. New York: McGraw-Hill, 2001.

[6] M. L. Honig and H. V. Poor, "Adaptive interference suppression," in *Wireless Communications: Signal Processing Perspectives*, H. V. Poor and G. W. Wornell, Eds. Englewood Cliffs, NJ: Prentice-Hall, 1998, ch. 2, pp. 64–128.

[7] H. Holma and A. Toskala, Eds., *WCDMA for UMTS—Radio Access for Third Generation Mobile Communications,* 3rd ed. New York: Wiley, 2004.

[8] M. L. Honig, S. L. Miller, M. J. Shensa, and L. B. Milstein, "Performance of adaptive linear interference suppression in the presence of dynamic fading," *IEEE Trans. Commun.*, vol. 49, no. 4, pp. 635–645, Apr. 2001.

[9] R. C. de Lamare and R. Sampaio-Neto, "Minimum mean squared error iterative successive parallel arbitrated decision feedback detectors for DS-CDMA systems," *IEEE Trans. Commun.*, vol. 56, no. 5, pp. 778–789, May 2008.

[10] *LAN/MAN Wireless LANS*, IEEE Std. 802.11, 2007.

[11] M. Z. Win and R. A. Scholtz, "Ultra-wide bandwidth time-hopping spread-spectrum impulse radio for wireless multiple access communications," *IEEE Trans. Commun.*, vol. 48, no. 4, pp. 679–691, Apr. 2000.

[12] M. Juntti, M. Vehkapera, J. Leinonen, V. Zexian, D. Tujkovic, S. Tsumura, and S. Hara, "MIMO MC-CDMA communications for future cellular systems," *IEEE Commun. Mag.*, vol. 43, no. 2, pp. 118–124, Feb. 2005.
[13] A. J. Paulraj, D. A. Gore, R. U. Nabar, and H. Bolcskei, "An overview of MIMO communications—A key to gigabit wireless," *Proc. IEEE*, vol. 92, no. 2, pp. 198–218, Feb. 2004.
[14] S. Haykin, *Adaptive Filter Theory,* 4th ed. Englewood Cliffs, NJ: Prentice-Hall, 2002.
[15] L. L. Scharf, "The SVD and reduced rank signal processing," *Signal Process.*, vol. 25, no. 2, pp. 113–133, Nov. 1991.
[16] L. L. Scharf and D. W. Tufts, "Rank reduction for modeling stationary signals," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. ASSP-35, no. 3, pp. 350–355, Mar. 1987.
[17] L. L. Scharf and B. van Veen, "Low rank detectors for Gaussian random vectors," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. ASSP-35, no. 11, pp. 1579–1582, Nov. 1987.
[18] A. M. Haimovich and Y. Bar-Ness, "An eigenanalysis interference canceler," *IEEE Trans. Signal Process.*, vol. 39, no. 1, pp. 76–84, Jan. 1991.
[19] X. Wang and H. V. Poor, "Blind multiuser detection: A subspace approach," *IEEE Trans. Inf. Theory*, vol. 44, no. 2, pp. 677–690, Mar. 1998.
[20] Y. Song and S. Roy, "Blind adaptive reduced-rank detection for DS-CDMA signals in multipath channels," *IEEE J. Sel. Areas Commun.*, vol. 17, no. 11, pp. 1960–1970, Nov. 1999.
[21] J. S. Goldstein and I. S. Reed, "Reduced rank adaptive filtering," *IEEE Trans. Signal Process.*, vol. 45, no. 2, pp. 492–496, Feb. 1997.
[22] Y. Hua and M. Nikpour, "Computing the reduced rank Wiener filter by IQMD," *IEEE Signal Process. Lett.*, vol. 6, no. 9, pp. 240–242, Sep. 1999.
[23] Y. Hua, M. Nikpour, and P. Stoica, "Optimal reduced rank estimation and filtering," *IEEE Trans. Signal Process.*, vol. 49, no. 3, pp. 457–469, Mar. 2001.
[24] 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.
[25] M. L. Honig and J. S. Goldstein, "Adaptive reduced-rank interference suppression based on the multistage Wiener filter," *IEEE Trans. Commun.*, vol. 50, no. 6, pp. 986–994, Jun. 2002.
[26] C. C. Hu and I. S. Reed, "Space–time adaptive reduced-rank multistage Wiener filtering for asynchronous DS-CDMA," *IEEE Trans. Signal Process.*, vol. 52, no. 7, pp. 1862–1877, Jul. 2004.
[27] S.-H. Wu, U. Mitra, and C.-C. J. Kuo, "Reduced-rank multistage receivers for DS-CDMA in frequency-selective fading channels," *IEEE Trans. Commun.*, vol. 53, no. 2, pp. 366–378, Feb. 2005.
[28] D. A. Pados and S. N. Batalama, "Low complexity blind detection of DS/CDMA signals: Auxiliary-vector receivers," *IEEE Trans. Commun.*, vol. 45, no. 12, pp. 1586–1594, Dec. 1997.
[29] 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, Sep. 1999.
[30] D. A. Pados, F. J. Lombardo, and S. N. Batalama, "Auxiliary vector filters and adaptive steering for DS-CDMA single-user detection," *IEEE Trans. Veh. Technol.*, vol. 48, no. 6, pp. 1831–1839, Nov. 1999.
[31] 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.
[32] H. Qian and S. N. Batalama, "Data record-based criteria for the selection of an auxiliary vector estimator of the MMSE/MVDR filter," *IEEE Trans. Commun.*, vol. 51, no. 10, pp. 1700–1708, Oct. 2003.
[33] W. Chen, U. Mitra, and P. Schniter, "On the equivalence of three reduced rank linear estimators with applications to DS-CDMA," *IEEE Trans. Inf. Theory*, vol. 48, no. 9, pp. 2609–2614, Sep. 2002.
[34] R. C. de Lamare and R. Sampaio-Neto, "Adaptive reduced-rank MMSE filtering with interpolated FIR filters and adaptive interpolators," *IEEE Signal Process. Lett.*, vol. 12, no. 3, pp. 177–180, Mar. 2005.
[35] R. C. de Lamare and R. Sampaio-Neto, "Reduced-rank interference suppression for DS-CDMA based on interpolated FIR filters," *IEEE Commun. Lett.*, vol. 9, no. 3, pp. 213–215, Mar. 2005.
[36] R. C. de Lamare and R. Sampaio-Neto, "Adaptive interference suppression for DS-CDMA systems based on interpolated FIR filters with adaptive interpolators in multipath channels," *IEEE Trans. Veh. Technol.*, vol. 56, no. 6, pp. 2457–2474, Sep. 2007.
[37] R. C. de Lamare and R. Sampaio-Neto, "Space–time adaptive reduced-rank multiuser receivers for DS-CDMA systems with interpolated FIR filters and time-varying interpolators," *IET Commun.*, vol. 2, no. 2, pp. 388–397, Feb. 2008.
[38] R. C. de Lamare and R. Sampaio-Neto, "Reduced-rank adaptive filtering based on joint iterative optimization of adaptive filters," *IEEE Signal Process. Lett.*, vol. 14, no. 12, pp. 980–983, Dec. 2007.
[39] R. C. de Lamare and R. Sampaio-Neto, "Adaptive space–time reduced-rank estimation based on diversity-combined decimation and interpolation applied to interference suppression in CDMA systems," *IET Signal Process.*, vol. 3, no. 2, pp. 150–163, Mar. 2009.
[40] 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.
[41] G. H. Golub and C. F. van Loan, *Matrix Computations,* 3rd ed. Baltimore, MD: Johns Hopkins Univ. Press, 1996.
[42] D. Luenberger, *Linear and Nonlinear Programming,* 2nd ed. Reading, MA: Addison-Wesley, 1984.
[43] T. S. Rappaport, *Wireless Communications*. Englewood Cliffs, NJ: Prentice-Hall, 1996.

Rodrigo C. de Lamare (S'99–M'04) received the B.S. degree in electronic engineering from the Federal University of Rio de Janeiro (UFRJ), Rio de Janeiro, Brazil, in 1998 and the M.Sc. and Ph.D. degrees in electrical engineering from the Pontifical Catholic University of Rio de Janeiro (PUC-Rio) in 2001 and 2004, respectively.

From January 2004 to June 2005, he was a Postdoctoral Fellow with the Center for Studies in Telecommunications, PUC-Rio. From July 2005 to January 2006, he was a Postdoctoral Fellow with the Signal Processing Laboratory, UFRJ. Since January 2006, he has been with the Communications Research Group, Department of Electronics, University of York, York, U.K., where he is currently a Lecturer in communications engineering. His research interests include communications and signal processing.

Raimundo Sampaio-Neto received the B.S. and M.Sc. degrees in electrical engineering from the Pontificia Universidade Católica do Rio de Janeiro (PUC-Rio), Rio de Janeiro, Brazil, in 1975 and 1978, respectively, and the Ph.D. degree in electrical engineering from the University of Southern California (USC), Los Angeles, in 1983.

From 1978 to 1979, he was an Assistant Professor with PUC-Rio. From 1979 to 1983, he was a Doctoral Student and a Research Assistant with the Department of Electrical Engineering, USC, with a fellowship from CAPES. From November 1983 to June 1984, he was a Postdoctoral Fellow with the Communication Sciences Institute, Department of Electrical Engineering, USC, and a Member of the technical staff with Axiomatic Corporation, Los Angeles. Since July 1984, he has been with PUC-Rio, where he is currently a Researcher with the Center for Studies in Telecommunications and an Associate Professor with the Department of Electrical Engineering. In 1991, he was a Visiting Professor with the Department of Electrical Engineering, USC. He has participated in various projects and has consulted for several private companies and government agencies. He is an Associate Editor of the *Journal of the Brazilian Communication Society*. His areas of interest include communication systems theory, digital transmission, satellite communications, and signal processing for communications.

Dr. Sampaio-Neto was a coorganizer of the Session on Recent Results for the IEEE Workshop on Information Theory in 1992 in Salvador, Brazil. He has also served as a Technical Program Cochairman for the IEEE Global Telecommunications Conference held in Rio de Janeiro in December 1999 and as a member of the technical program committees of several national and international conferences. He was in office for two consecutive terms on the Board of Directors of the Brazilian Communications Society, where he is currently a member of its Advisory Council.