Adaptive Reduced-Rank Processing Based on Joint and Iterative Interpolation, Decimation and Filtering

Rodrigo C. de Lamare and Raimundo Sampaio-Neto

Abstract—**We present an adaptive reduced-rank signal processing technique for performing dimensionality reduction in general adaptive filtering problems. The proposed method is based on the concept of joint and iterative interpolation, decimation and filtering. We describe an iterative least squares (LS) procedure to jointly optimize the interpolation, decimation and filtering tasks for reduced-rank adaptive filtering. In order to design the decimation unit, we present the optimal decimation scheme based on the counting principle and also propose low-complexity decimation structures. We then develop low-complexity least-mean squares (LMS) and recursive least squares (RLS) algorithms for the proposed scheme along with automatic rank and branch adaptation techniques. An analysis of the convergence properties and issues of the proposed algorithms is carried out and the key features of the optimization problem such as the existence of multiple solutions are discussed. We consider the application of the proposed algorithms to interference suppression in CDMA systems. Simulations results show that the proposed algorithms outperform the best known reduced-rank schemes with lower complexity.**

Keywords—**Adaptive filtering, reduced-rank techniques, adaptive algorithms, CDMA systems .**

I. Introduction

IN the literature of adaptive filtering algorithms [1], [2], a large number of algorithms with different trade-offs between performance and complexity has been reported. In general, the convergence and tracking performances of these algorithms depend on the eigenvalue spread of the full-rank covariance matrix $\mathbf{R}$ of the input data $\mathbf{r}(i)$ and the number of elements M in the filter [1], [2]. A challenging problem is to perform signal processing when the number of elements in the filter is very large and the algorithm requires a large number of samples (or data record) to be trained. Furthermore, in highly dynamic systems such as those in wireless communications, adaptive filters with a large number of elements usually fail or provide poor performance in tracking signals embedded in interference and noise.

Reduced-rank filtering is a very powerful technique that has gained considerable attention in the last few years due to its effectiveness in low-sample-support situations where it can offer improved convergence performance at an affordable complexity [6]-[22]. The origins of reduced-rank filtering lie in the problem of feature selection encountered in statistical signal processing [6]. The aim of these methods is to devise a transformation in such a way that the data vector can be represented by a reduced number of effective features and yet retain most of the intrinsic information content of the input data [6]. In this context, the existing reduced-rank methods obtain a low-rank approximation of $\mathbf{r}(i)$ with dimension M that provides faster acquisition of the signal statistics usually leading to better convergence and tracking performance. Among the available reduced-rank filtering methods are the eigen-decomposition techniques [6]-[11], the multistage Wiener filter (MWF) [12],[13], the auxiliary-vector filtering (AVF) algorithm [17]-[21], the adaptive interpolated FIR filters with time-varying interpolators [22], [23] that are rank limited and the joint iterative reduced-rank scheme based on optimization of a projection matrix and a reduced-rank filter [24]. The major problem with the eigen-decomposition, the MWF, the AVF and the method in [24] is that they rely on estimates of $\mathbf{R}$ as a starting point for the subspace decomposition. The estimation process of $\mathbf{R}$ with time averages can be problematic, often requires large data records, and may experience tracking problems in dynamic situations.

In this work, we propose a reduced-rank adaptive filtering scheme based on a joint and iterative procedure that optimizes the interpolation, the decimation and the filtering tasks. In this scheme, which was firstly presented in [25], [26], the number of elements for filtering is substantially reduced, resulting in considerable computational savings and very fast convergence performance for general filtering applications. A unique feature of the proposed method is that, unlike existing schemes, it does not rely on estimates of $\mathbf{R}$ (that requires at least $2M$ data vectors for convergence in non-stationary scenarios [1], [2]) before projecting the received data onto a reduced-rank subspace. The proposed approach skips the estimation of $\mathbf{R}$ and obtains directly the subspace of interest via a set of simple interpolation and decimation operations. We describe the optimal decimation scheme and low-complexity decimation schemes for the proposed structure. We derive LMS and RLS algorithms for implementing the proposed scheme and evaluate their computational complexity. We also propose rank and branch adaptation techniques for automatically determining the number of branches necessary to achieve a pre-specified performance and the best rank for the interpolator and reduced-rank filters. The convergence issues and properties of the method are discussed and several important features of the optimization problem such as the existence of multiple solutions are discussed. We consider the application of the proposed scheme to interference suppression in CDMA systems.

The rest of this work is organized as follows. Section II states the problem and discusses the design of reduced-rank filters. Section III presents the proposed reduced-rank adaptive filtering scheme, describes the proposed joint iterative least squares (LS) optimization of the interpolation, decimation and filtering tasks, and details the proposed decimation schemes. In Section IV, we present LMS and RLS algorithms for the joint optimization of the interpolator, decimation unit and the reduced-rank fil-

EDICS: ASP-ANAL, ASP-APPL, SPC-SPRD. Dr. R. C. de Lamare is with the University of York, York Y010 5DD, UK and Prof. R. Sampaio-Neto is with CETUC/PUC-RIO, 22453-900, Rio de Janeiro, Brazil. E-mails: rcdl500@ohm.york.ac.uk, raimundo@cetuc.puc-rio.br. This work was supported in part by the UK Ministry of Defence, contract MOD-COI RT/COM/5/021.
USPTO Application No. 11/427.471 - Patent Pending.

ters along with the rank and branch adaptation techniques. The convergence properties of the proposed method and the existence of solutions are studied in Section V. Section VI presents and discusses the simulation results, while Section VII gives the conclusions.

II. Design of Reduced-Rank Filters and Problem Statement

Let us first consider a general reduced-rank filtering design problem. A reduced-rank algorithm shall extract the most important features of the processed data and reduce the number of parameters for estimation. This dimensionality reduction is accomplished by projecting the received vector onto a lower dimensional subspace. Specifically, consider an $M \times D$ projection matrix $\mathbf{S}_D$ which carries out a dimensionality reduction on the $M \times 1$ input data vector $\mathbf{r}(i)$ as given by

$$\bar{\mathbf{r}}(i) = \mathbf{S}_D^H \mathbf{r}(i) \tag{1}$$

where, in what follows, all D-dimensional quantities are denoted with a "bar". The resulting projected received vector $\bar{\mathbf{r}}(i)$ is the input to a filter represented by the $D \times 1$ vector $\bar{\mathbf{w}} = [\bar{w}_1 \ \bar{w}_2 \ \ldots \bar{w}_D]^T$. The filter output corresponding to the ith time instant is

$$x(i) = \bar{\mathbf{w}}^H \bar{\mathbf{r}}(i) \tag{2}$$

Let us perform the MMSE filter design and substitute (2) into the MSE cost function

$$\mathcal{J} = E\big[|d(i) - x(i)|^2\big] \tag{3}$$

where $d(i)$ is the desired signal. The reduced-rank filter that solves (3) is given by

$$\bar{\mathbf{w}} = \bar{\mathbf{R}}^{-1}\bar{\mathbf{p}} \tag{4}$$

where $\bar{\mathbf{R}} = E[\bar{\mathbf{r}}(i)\bar{\mathbf{r}}^H(i)] = \mathbf{S}_D^H \mathbf{R} \mathbf{S}_D$ is the reduced-rank covariance matrix, $\bar{\mathbf{p}} = E[d^*(i)\bar{\mathbf{r}}(i)] = \mathbf{S}_D^H \mathbf{p}$ is the reduced-rank cross-correlation vector and $\mathbf{p} = E[d^*(i)\mathbf{r}(i)]$ is the cross-correlation vector. The associated MMSE for a rank-D filter is expressed by

$$\text{MMSE} = \sigma_d^2 - \bar{\mathbf{p}}^H \bar{\mathbf{R}}^{-1}\bar{\mathbf{p}} = \sigma_b^2 - \mathbf{p}^H \mathbf{S}_D (\mathbf{S}_D^H \mathbf{R} \mathbf{S}_D)^{-1} \mathbf{S}_D^H \mathbf{p} \tag{5}$$

where σ_d^2 is the variance of $d(i)$. Based upon the problem statement above, the design of reduced-rank schemes can be stated as follows. How to efficiently (or optimally) design a transformation matrix $\mathbf{S}_D$ that projects the input data vector $\mathbf{r}(i)$ onto a reduced-rank data vector $\bar{\mathbf{r}}(i)$? This problem is addressed by proposed adaptive reduced-rank filtering scheme detailed next.

III. Proposed Adaptive Reduced-Rank Filtering Scheme

In this section we detail the proposed adaptive reduced-rank filtering scheme based on the joint and iterative interpolation, decimation and filtering. The motivation for designing a transformation $\mathbf{S}_D(i)$ based on interpolation and decimation comes from two observations. The first is that rank reduction can be performed by eliminating (decimating) samples that are not useful in the filtering process and then attempting to recreate the eliminated samples with an interpolator. The second comes from the structure of $\mathbf{S}_D(i)$, whose columns can be represented by a time-varying interpolator and a time-varying decimator and which can form bases for dimensionality reduction. The proposed scheme, denoted JIDF, is shown in Fig. 1, where an interpolator, time-varying decimation unit and a reduced-rank filter that are time-varying are employed. The input vector $\mathbf{r}(i) = [r_0^{(i)} \ \ldots \ r_{M-1}^{(i)}]^T$ is filtered by the interpolator filter $\mathbf{v}(i) = [v_0^{(i)} \ \ldots \ v_{N_I-1}^{(i)}]^T$ and yields the interpolated vector $\mathbf{r}_\text{I}(i)$ with M samples, which is expressed by

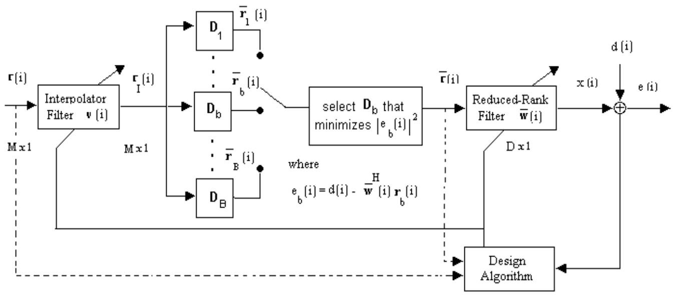

Fig. 1. Proposed Adaptive Reduced-Rank Filtering Scheme.

$$\mathbf{r}_\text{I}(i) = \mathbf{V}^H(i)\mathbf{r}(i) \tag{6}$$

where the $M \times M$ Toeplitz convolution matrix $\mathbf{V}(i)$ is given by

$$\mathbf{V}(i) = \begin{bmatrix} v_0^{(i)} & 0 & \ldots & 0 \\ \vdots & v_0^{(i)} & \ldots & 0 \\ v_{N_I-1}^{(i)} & \vdots & \ldots & 0 \\ 0 & v_{N_I-1}^{(i)} & \ldots & 0 \\ 0 & 0 & \ddots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \ldots & v_0^{(i)} \end{bmatrix}.$$

In order to facilitate the description of the scheme, let us introduce an alternative way of expressing the vector $\mathbf{r}_\text{I}(i)$, that will be useful in the following through the equivalence:

$$\mathbf{r}_\text{I}(i) = \mathbf{V}^H(i)\mathbf{r}(i) = \boldsymbol{\Re}_o(i)\mathbf{v}^*(i), \tag{7}$$

where the $M \times N_I$ matrix $\boldsymbol{\Re}_o(i)$ with the samples of $\mathbf{r}(i)$ has a Hankel structure [27] and is described by

$$\boldsymbol{\Re}_o(i) = \begin{bmatrix} r_0^{(i)} & r_1^{(i)} & \ldots & r_{N_I-1}^{(i)} \\ r_1^{(i)} & r_2^{(i)} & \ldots & r_{N_I}^{(i)} \\ \vdots & \vdots & \ddots & \vdots \\ r_{M-1}^{(i)} & r_M^{(i)} & \ldots & r_{M+N_I-2}^{(i)} \end{bmatrix}. \tag{8}$$

The dimensionality reduction is performed by a decimation unit with $D \times M$ decimation matrices $\mathbf{D}_b(i)$ that projects $\mathbf{r}_\text{I}(i)$ onto $D \times 1$ vectors $\bar{\mathbf{r}}_b(i)$ with $b = 1, \ldots, B$, where $D = M/L$

is the rank and L is the decimation factor. The $D \times 1$ vector $\bar{\mathbf{r}}_b(i)$ for branch b is expressed by

$$\bar{\mathbf{r}}_b(i) = \mathbf{D}_b(i)\mathbf{r}_{\mathrm{I}}(i) = \mathbf{D}_b(i)\boldsymbol{\Re}_o(i)\mathbf{v}^*(i), \quad (9)$$

where the vector $\bar{\mathbf{r}}_b(i)$ for branch b is used in the minimization of the squared norm of the error for branch b

$$e_b(i) = d(i) - \bar{\mathbf{w}}^H(i)\bar{\mathbf{r}}_b(i).$$

The decimation pattern $\mathbf{D}(i)$ is selected according to:

$$\mathbf{D}(i) = \mathbf{D}_{b_s}(i) \text{ when } b_s = \arg \min_{1\leq b\leq B} |e_b(i)|^2, \quad (10)$$

where B is the number of decimation branches which is a parameter to be set by the designer.

After the decimation unit, which carries out dimensionality reduction, the proposed JIDF scheme employs a reduced-rank FIR filter $\bar{\mathbf{w}}(i)$ with D elements to yield the output of the scheme. A key strategy for the joint and iterative optimization that follows is to express the output of the JIDF structure $x(i) = \bar{\mathbf{w}}^H(i)\bar{\mathbf{r}}(i)$ as a function of $\mathbf{v}(i)$, the decimation matrix $\mathbf{D}(i)$ and $\bar{\mathbf{w}}(i)$ as follows:

$$x(i) = \bar{\mathbf{w}}^H(i)\mathbf{S}_D^H(i)\mathbf{r}(i) = \bar{\mathbf{w}}^H(i)\mathbf{D}(i)\boldsymbol{\Re}_o(i)\mathbf{v}^*(i) = \mathbf{v}^H(i)\mathbf{u}(i), \quad (11)$$

where $\mathbf{u}(i) = \boldsymbol{\Re}_o^T(i)\mathbf{D}^T(i)\bar{\mathbf{w}}^*(i)$ is an $N_I \times 1$ vector. The expression in (11) indicates that the dimensionality reduction carried out by the proposed scheme depends on finding appropriate $\mathbf{v}(i)$, $\mathbf{D}(i)$ for constructing $\mathbf{S}_D(i)$ as shown next.

A. Joint and Iterative Least Squares Optimization Algorithm

We will describe in this part the proposed joint and iterative optimization algorithm that adjusts the parameters of the interpolator filter $\mathbf{v}(i)$, selects the decimation pattern $\mathbf{D}(i)$ and adjusts the reduced-rank filter $\bar{\mathbf{w}}(i)$. The proposed optimization considers the exponentially weighted LS cost function

$$\begin{aligned}\mathcal{J}_{LS}^{(\mathbf{v}(i),\mathbf{D}(i),\bar{\mathbf{w}}(i))} &= \sum_{l=1}^{i} \alpha^{i-l}|\ d(l) - \bar{\mathbf{w}}^H(i)\mathbf{S}_D^H(i)\mathbf{r}(l)\ |^2 \\ &= \sum_{l=1}^{i} \alpha^{i-l}|\ d(l) - \mathbf{v}^H(i)\boldsymbol{\Re}_o^T(l)\mathbf{D}^T(i)\bar{\mathbf{w}}^*(i)\ |^2\end{aligned} \quad (12)$$

The decimation pattern $\mathbf{D}(i)$ is optimized according to the a priori error at branch b

$$\xi_b(i) = d(i) - \bar{\mathbf{w}}^H(i-1)\mathbf{D}_b(i)\boldsymbol{\Re}_o(i)\mathbf{v}^*(i-1) \quad (13)$$

and selected on the basis of the following criterion:

$$\mathbf{D}(i) = \mathbf{D}_{b_{\mathrm{opt}}}(i) \text{ when } b_{\mathrm{opt}} = \arg \min_{1\leq b\leq B} |\xi_b(i)|^2, \quad (14)$$

By fixing $\mathbf{D}(i)$ and $\bar{\mathbf{w}}(i)$, taking the gradient of (12) with respect to $\mathbf{v}(i)$, equating it to a null vector and solving the equation, the interpolator $\mathbf{v}(i)$ that minimizes (12) is expressed by

$$\mathbf{v}(i) = \boldsymbol{\beta}(\bar{\mathbf{w}}(i-1), \mathbf{D}(i)) = \bar{\mathbf{R}}_{\mathbf{u}}^{-1}(i)\bar{\mathbf{p}}_{\mathbf{u}}(i), \quad (15)$$

where $\mathbf{u}(i) = \boldsymbol{\Re}_o^T(i)\mathbf{D}^T(i)\bar{\mathbf{w}}^*(i-1)$, $\bar{\mathbf{p}}_{\mathbf{u}}(i) = \sum_{l=1}^{i}\alpha^{i-l}d^*(l)\mathbf{u}(l)$ and $\bar{\mathbf{R}}_{\mathbf{u}}(i) = \sum_{l=1}^{i}\alpha^{i-l}\mathbf{u}(l)\mathbf{u}^H(l)$.

By fixing $\mathbf{v}(i)$ and $\mathbf{D}(i)$, taking the gradient of (12) with respect to $\bar{\mathbf{w}}(i)$, equating it to a null vector and solving the equation, the reduced-rank filter that minimizes (12) is given by

$$\bar{\mathbf{w}}(i) = \boldsymbol{\gamma}(\mathbf{v}(i), \mathbf{D}(i)) = \bar{\mathbf{R}}^{-1}(i)\bar{\mathbf{p}}(i), \quad (16)$$

where $\bar{\mathbf{r}}(i) = \mathbf{D}(i)\boldsymbol{\Re}_o(i)\mathbf{v}^*(i)$, $\bar{\mathbf{p}}(i) = \sum_{l=1}^{i}\alpha^{i-l}d^*(l)\bar{\mathbf{r}}(l)$ and $\bar{\mathbf{R}}(i) = \sum_{l=1}^{i}\alpha^{i-l}\bar{\mathbf{r}}(l)\bar{\mathbf{r}}^H(l)$. The associated sum of error squares (SES) expressions are given by

$$\mathcal{J}(\mathbf{v}(i)) = \mathcal{J}_{LS}(\mathbf{v}(i), \mathbf{D}(i), \boldsymbol{\gamma}(\mathbf{v}(i), \mathbf{D}(i)))) = \varepsilon_d - \bar{\mathbf{p}}^H(i)\bar{\mathbf{R}}^{-1}(i)\bar{\mathbf{p}}(i), \quad (17)$$

$$\mathcal{J}_{LS}(\boldsymbol{\beta}(\bar{\mathbf{w}}(i), \mathbf{D}(i)), \mathbf{D}(i), \bar{\mathbf{w}}(i)) = \varepsilon_d - \mathbf{p}_{\mathbf{u}}^H(i)\mathbf{R}_{\mathbf{u}}^{-1}(i)\mathbf{p}_{\mathbf{u}}(i), \quad (18)$$

where $\varepsilon_d = \sum_{l=1}^{i}\alpha^{i-l}|d(l)|^2$ is the energy of the desired response. This structure trades off a full-rank matrix inversion against the inversion of two matrices with rank D and N_I and a branch selection procedure. Note that the expressions (14), (15) and (16) are not closed-form solutions since they depend on each other. Therefore, it is necessary to iterate (14), (15) and (16) with an initial guess to obtain a solution for the proposed LS optimization. Low-complexity joint and iterative solutions are sought via adaptive algorithms in the next section.

B. Design of the Decimation Unit

In this subsection, we present design strategies for the decimation unit. In particular, we consider the optimal approach and three alternative procedures for designing the decimation unit of the novel reduced-rank filtering scheme, where the common framework is the use of parallel branches with B switching decimation patterns. The design of $\mathbf{D}(i)$ is constrained such that the elements of the matrix are either 0s or 1s. This means that in our proposed approach the decimation unit simply keeps or discards samples, leading to a very simple implementation.

The exhaustive decimation pattern $\mathbf{D}_{\mathrm{opt}}$ for the proposed scheme can be obtained considering all possible design patterns. It selects the optimal pattern according to

$$b_{\mathrm{opt}} = \arg \min_{1\leq b\leq B} |\xi_b(i)|^2, \quad (19)$$

where an exhaustive procedure that selects D samples out of M possible candidates is performed. The total number of patterns B is equal to

$$B = \underbrace{M\cdot(M-1)\ldots(M-D+1)}_{D \text{ terms}} = \frac{M!}{(M-D)!} = \begin{pmatrix} M \\ D \end{pmatrix}.$$

We can view this procedure as a combinatorial problem that has M samples as possible candidates for the first row of $\mathbf{D}_{\mathrm{opt}}$ and $M-m+1$ samples as candidates for the following $D-1$ rows of $\mathbf{D}_{\mathrm{opt}}$ is considered. The quantity m denotes the mth row of the matrix $\mathbf{D}_{\mathrm{opt}}$. The exhaustive decimation scheme described above is, however, too complex for practical use because it requires D permutations of M samples for each symbol interval and carries out an extensive search over all possible patterns. Therefore, a decimation scheme that renders itself to practical and low-complexity implementations is of great interest.

To this end, we consider a general framework for sub-optimal decimation schemes with rank $D = M/L$, decimation factor L and B parallel branches as described by the structure:

$$\mathbf{D}_b = \begin{bmatrix} \underbrace{0 \ \dots \ 0}_{r_1 \ zeros} & 1 & 0 & 0 & 0 & 0 & \dots & 0 & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ \underbrace{0 \ 0 \ \dots \ 0}_{r_m \ zeros} & 1 & 0 & 0 & 0 & \dots & 0 & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ \underbrace{0 \ 0 \ 0 \ 0 \ \ 0 \ 0 \ \ 0 \dots 0}_{r_D \ zeros} & 1 & \underbrace{0 \ \dots \ 0}_{(M-r_D-1) \ zeros} \end{bmatrix}, \quad (20)$$

where m ($m = 1, 2, \dots, D$) denotes the m-th row and r_m is the number of zeros chosen according to the following proposed alternative decimation patterns:

A. Uniform (U) Decimation with $B = 1$. We make $r_m = (m-1)L$ and this corresponds to the use of a single branch ($B = 1$) on the decimation unit (no switching and optimization of branches), and is equivalent to the scheme reported in [23].

B. Pre-Stored (PS) Decimation. We select $r_m = (m-1)L + (b-1)$ which corresponds to the utilization of uniform decimation for each branch b out of B branches and the different patterns are obtained by picking out adjacent samples with respect to the previous and succeeding decimation patterns.

C. Random (R) Decimation. We choose r_m as a discrete uniform random variable, which is independent for each row m out of B branches and whose values range between 0 and $M-1$.

IV. Adaptive Algorithms

In this section we describe LMS and RLS algorithms [1] that jointly and iteratively estimate the parameters of the interpolator filter, the decimation pattern and the reduced-rank filter of the proposed scheme. The computational complexity of the proposed scheme equipped with the LMS and the RLS algorithms is detailed and compared with existing methods. We also present automatic rank adaptation algorithms for adjusting the ranks/lengths of $\mathbf{v}(i)$ and $\mathbf{w}(i)$, and an algorithm for determining the minimum number of branches necessary to achieve a pre-specified performance.

A. LMS algorithms

In order to derive LMS algorithms for the proposed scheme, we consider the following cost function

$$J_{\rm MSE}^{(\mathbf{v}(i),\mathbf{D}(i),\bar{\mathbf{w}}(i))} = E[|d(i) - \mathbf{v}^H(i)\boldsymbol{\Re}_o^T(i)\mathbf{D}^T(i)\bar{\mathbf{w}}^*(i)|^2]. \quad (21)$$

The decimation pattern $\mathbf{D}(i)$ is optimized according to the instantaneous error at branch b

$$e_b(i) = d(i) - \bar{\mathbf{w}}^H(i)\mathbf{D}_b(i)\boldsymbol{\Re}_o(i)\mathbf{v}^*(i), \quad (22)$$

and the optimal branch is selected according to:

$$b_{\rm opt} = \arg\min_{1 \le b \le B} |e_b(i)|^2, \quad (23)$$

Then we employ the following quantities for the adaptation of $\mathbf{v}(i)$ and $\bar{\mathbf{w}}(i)$:

$$\mathbf{D}(i) = \mathbf{D}_{b_{\rm opt}}(i), \ \ \bar{\mathbf{r}}(i) = \bar{\mathbf{r}}_{b_{\rm opt}}(i) \ \text{ and } \ e(i) = e_{b_{\rm opt}}(i) \quad (24)$$

where $\bar{\mathbf{r}}(i) = \mathbf{D}(i)\boldsymbol{\Re}_o(i)\mathbf{v}^*(i)$. Taking the instantaneous gradient terms of (21) with respect to $\mathbf{v}(i)$ and using the gradient descent rules [1], [2] for the interpolator $\mathbf{v}(i+1) = \mathbf{v}(i) - \eta\nabla_{\mathbf{v}} J_{\rm MSE}^{(\mathbf{v}(i),\mathbf{D}(i),\bar{\mathbf{w}}(i))}$, we get the update rule for $\mathbf{v}(i)$

$$\mathbf{v}(i+1) = \mathbf{v}(i) + \eta e^*(i)\mathbf{u}_p(i), \quad (25)$$

where $\mathbf{u}_p(i) = \boldsymbol{\Re}_o^T(i)\mathbf{D}^T(i)\bar{\mathbf{w}}^*(i)$. Taking the instantaneous gradient terms of (21) with respect to $\bar{\mathbf{w}}(i)$ and using the gradient descent rules [1], [2] $\bar{\mathbf{w}}(i+1) = \bar{\mathbf{w}}(i) - \mu\nabla_{\mathbf{w}} J_{\rm MSE}^{(\mathbf{v}(i),\mathbf{D}(i),\bar{\mathbf{w}}(i))}$ yields the update rule for $\bar{\mathbf{w}}(i)$:

$$\bar{\mathbf{w}}(i+1) = \bar{\mathbf{w}}(i) + \mu e^*(i)\bar{\mathbf{r}}(i), \quad (26)$$

where $e(i) = d(i) - \bar{\mathbf{w}}^H(i)\bar{\mathbf{r}}(i)$, the quantities μ and η are the step sizes for $\bar{\mathbf{w}}(i)$ and $\mathbf{v}(i)$, respectively. The LMS algorithms for the proposed structure described in this section have a computational complexity $O(D + N_I)$. In fact, the proposed structure trades off one LMS algorithm with complexity $O(M)$ against two LMS algorithms with complexity $O(D)$ and $O(N_I)$, operating simultaneously with a switching decimation scheme.

B. RLS algorithms

In order to derive RLS algorithms for the proposed scheme we consider the LS cost function given in (12) and follow a similar approach to the proposed LMS algorithms. The decimation pattern $\mathbf{D}(i)$ is chosen according to the a priori error at branch b

$$\xi_b(i) = d(i) - \bar{\mathbf{w}}^H(i-1)\mathbf{D}_b(i)\boldsymbol{\Re}_o(i)\mathbf{v}^*(i-1), \quad (27)$$

and the optimal branch is selected according to:

$$b_{\rm opt} = \arg\min_{1 \le b \le B} |\xi_b(i)|^2, \quad (28)$$

Then we employ the following quantities for the adaptation of $\mathbf{v}(i)$ and $\bar{\mathbf{w}}(i)$:

$$\mathbf{D}(i) = \mathbf{D}_{b_{\rm opt}}(i), \ \ \bar{\mathbf{r}}(i) = \bar{\mathbf{r}}_{b_{\rm opt}}(i) \ \text{ and } \ \xi(i) = \xi_{b_{\rm opt}}(i) \quad (29)$$

In order to compute parameter estimates and avoid the inversion of $\bar{\mathbf{R}}(i)$ required in (15), we use the matrix inversion lemma (MIL) [1], define $\mathbf{P}_{\mathbf{u}}(i) = \bar{\mathbf{R}}_{\mathbf{u}}^{-1}(i)$ and the gain vector $\mathbf{G}_{\mathbf{u}}(i)$ as:

$$\mathbf{G}_{\mathbf{u}}(i) = \frac{\alpha^{-1}\mathbf{P}_{\mathbf{u}}(i-1)\mathbf{u}(i)}{1 + \alpha^{-1}\mathbf{u}^H(i)\mathbf{P}_{\mathbf{u}}(i-1)\mathbf{u}(i)}, \quad (30)$$

and thus we can rewrite $\mathbf{P}_{\mathbf{u}}(i)$ as

$$\mathbf{P}_{\mathbf{u}}(i) = \alpha^{-1}\mathbf{P}_{\mathbf{u}}(i-1) - \alpha^{-1}\mathbf{G}_{\mathbf{u}}(i)\mathbf{u}^H(i)\mathbf{P}_{\mathbf{u}}(i-1) \quad (31)$$

By rearranging (30) we have $\mathbf{G}_{\mathbf{u}}(i) = \alpha^{-1}\mathbf{P}_{\mathbf{u}}(i-1)\mathbf{u}_k(i) - \alpha^{-1}\mathbf{G}_{\mathbf{u}}(i)\mathbf{u}^H(i)\mathbf{P}_{\mathbf{u}}(i-1)\mathbf{u}(i) = \mathbf{P}_{\mathbf{u}}(i)\mathbf{u}(i)$. Using the LS solution in (12) and the recursion $\mathbf{p}_{\mathbf{u}}(i) = \alpha\mathbf{p}_{\mathbf{u}}(i-1) + \mathbf{u}(i)d^*(i)$ we arrive at

$$\mathbf{v}(i) = \mathbf{v}(i-1) + \mathbf{G}_{\mathbf{v}}(i)\xi^*(i) \quad (32)$$

where the *a priori* estimation error is described by $\xi(i) = d(i) - \mathbf{v}^H(i-1)\mathbf{u}(i) = d(i) - \bar{\mathbf{w}}^H(i-1)\bar{\mathbf{r}}(i)$. Similar recursions for the reduced-rank filter $\bar{\mathbf{w}}(i)$ can be devised by using (16).

To avoid the inversion of $\bar{\mathbf{R}}(i)$ we use the MIL again, define $\mathbf{P}(i) = \bar{\mathbf{R}}^{-1}(i)$ and the gain vector $\mathbf{G}(i)$:

$$\mathbf{G}(i) = \frac{\alpha^{-1}\mathbf{P}(i-1)\bar{\mathbf{r}}(i)}{1+\alpha^{-1}\bar{\mathbf{r}}^H(i)\mathbf{P}(i-1)\bar{\mathbf{r}}(i)}, \quad (33)$$

and thus we can rewrite $\mathbf{P}(i)$ as

$$\mathbf{P}(i) = \alpha^{-1}\mathbf{P}(i-1) - \alpha^{-1}\mathbf{G}(i)\bar{\mathbf{r}}^H(i)\mathbf{P}(i-1) \quad (34)$$

By rearranging (33) we have $\mathbf{G}(i) = \alpha^{-1}\mathbf{P}(i-1)\bar{\mathbf{r}}(i) - \alpha^{-1}\mathbf{G}(i)\bar{\mathbf{r}}^H(i)\mathbf{P}(i-1)\bar{\mathbf{r}}(i) = \mathbf{P}(i)\bar{\mathbf{r}}(i)$. Using the LS solution in (12) and the recursion $\mathbf{p}(i) = \alpha\mathbf{p}(i-1) + \bar{\mathbf{r}}(i)d^*(i)$ we obtain

$$\bar{\mathbf{w}}(i) = \bar{\mathbf{R}}^{-1}(i)\hat{\mathbf{p}}(i) = \alpha\mathbf{P}(i)\mathbf{p}(i-1) + \mathbf{P}(i)\bar{\mathbf{r}}(i)d^*(i). \quad (35)$$

Substituting (34) into (35) yields:

$$\bar{\mathbf{w}}(i) = \bar{\mathbf{w}}(i-1) + \mathbf{G}(i)\xi^*(i), \quad (36)$$

The RLS algorithm for the proposed scheme trades off a computational complexity of $\mathcal{O}((M)^2)$ against two RLS algorithms operating simultaneously with a switching decimation scheme, with complexity $\mathcal{O}((D)^2)$ and $\mathcal{O}(N_I^2)$, respectively, with D and $N_I << M$, as will be explained in the subsequent sections.

C. Computational Complexity

Here we illustrate the computational complexity of the proposed structure and algorithms. In Table I we show the computational complexity required by the proposed and existing algorithms. The proposed JIDF scheme with the LMS trades off a computational complexity of $\mathcal{O}((M))$ required by the full-rank LMS against two LMS algorithms operating simultaneously with complexities $\mathcal{O}((D))$ and $\mathcal{O}(N_I)$. With respect to the RLS, the JIDF trades off the complexity $\mathcal{O}((M)^2)$ required by the full-rank RLS against two RLS algorithms operating simultaneously with complexity $\mathcal{O}((D)^2)$ and $\mathcal{O}(N_I^2)$. If the designer chooses a small N_I and B and the decimation factor L sufficiently large then the complexity can be greatly reduced as the filter's rank $D = M/L$ is inversely proportional to L. The MWF technique has a complexity $\mathcal{O}(D\bar{M}^2)$, where the variable dimension of the vectors $\bar{M} = M - d$ varies according to the orthogonal decomposition and the rank $d = 1, \ldots, D$. The AVF method with non-orthogonal auxiliary vectors [20] has a complexity $\mathcal{O}((DM)^2)$.

In Fig. 2 we show curves which describe the computational complexity in terms of the arithmetic operations (additions and multiplications) as a function of the number of received samples M. The curves indicate that a significant computational advantage of the proposed scheme over the full-rank design is verified for the RLS algorithms. In comparison with the existing MWF and AVF reduced-rank techniques, the proposed JIDF scheme is substantially less complex.

D. Automatic Rank Adaptation of Filters

The performance of the algorithms described in the previous subsections depends on the ranks D and N_I. Unlike prior methods for rank selection which utilize MWF-based algorithms [13] or AVF-based recursions [21], we focus on an approach that jointly determines based on the LS criterion the lengths of the two filters $\mathbf{v}(i)$ and $\bar{\mathbf{w}}(i)$, namely, N_I and D. In particular, we present a method for automatically selecting the ranks of the algorithms based on the exponentially weighed *a posteriori* least-squares type cost function described by

$$\mathcal{C}(\bar{\mathbf{w}}^{(D)}, \mathbf{v}^{(N_I)}, \mathbf{D}) = \sum_{l=1}^{i} \alpha^{i-l} \big| d(l) - \bar{\mathbf{w}}^{H,\,(D)}(l)\mathbf{D}(l)\boldsymbol{\Re}_o(l)\mathbf{v}^{*,\,(N_I)}(l)\big|^2 \quad (37)$$

where α is the forgetting factor, $\tilde{\mathbf{w}}^{(D)}(i-1)$ is the reduced-rank filter with rank D and $\mathbf{v}^{(N_I)}(i)$ is the interpolator filter with rank N_I. For each time interval i and a given decimation pattern and B, we can select D and N_I which minimizes $\mathcal{C}(\bar{\mathbf{w}}^{(D)}, \mathbf{v}^{(N_I)}, \mathbf{D})$ and the exponential weighting factor α is required as the optimal rank varies as a function of the data record. The proposed rank adaptation algorithm that chooses the best lengths $D_{\rm opt}$ and $N_{I_{\rm opt}}$ for the filters $\mathbf{v}(i)$ and $\bar{\mathbf{w}}(i)$,

TABLE I

Computational complexity of algorithms.

	Number of operations per symbol	
Algorithm	Additions	Multiplications
Full-rank-LMS	$2M$	$2M+1$
Full-rank-RLS	$3(M-1)^2+M^2+2M$	$6M^2+2M+2$
JIDF-LMS	$(B+1)(D)+2N_I$	$(B+2)D$
JIDF-RLS	$3(D-1)^2+3(N_I-1)^2$ $+(D-1)N_I+N_IM+(D)^2$ $+N_I^2+(B+1)D+2N_I$	$6(D)^2+6N_I^2$ $+DN_I+2$ $+(B+2)D+N_I$
MWF-LMS	$D(2(\bar{M}-1)^2+\bar{M}+3)$	$D(2\bar{M}^2+5\bar{M}+7)$
MWF-RLS	$D(4(\bar{M}-1)^2+2\bar{M})$	$D(4\bar{M}^2+2\bar{M}+3)$
AVF	$D((M)^2+3(M-1)^2)-1$ $+D(5(M-1)+1)+2M$	$D(4(M)^2+4M+1)$ $+4M+2$

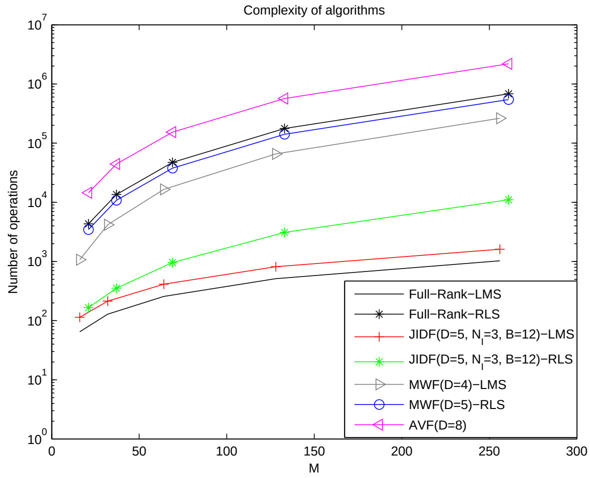

Fig. 2. Complexity in terms of arithmetic operations against number of received samples (M).

respectively, is given by

$$\{D_{\rm opt}, N_{I_{\rm opt}}\} = \arg \min_{\substack{N_{I_{\rm min}} \leq n \leq N_{I_{\rm max}} \\ D_{\rm min} \leq d \leq D_{\rm max}}} \mathcal{C}(\bar{\mathbf{w}}^{(d)}, \mathbf{v}^{(n)}, \mathbf{D}) \quad (38)$$

where d and n are integers, $D_{\rm min}$ and $D_{\rm max}$, and $N_{I_{\rm min}}$ and $N_{I_{\rm max}}$ are the minimum and maximum ranks allowed for the reduced-rank filter and the interpolator, respectively. Note that a smaller rank may provide faster adaptation during the initial stages of the estimation procedure and a slightly greater rank usually yields a better steady-state performance. Our studies reveal that the range for which the ranks D and N_I of the proposed algorithms have a positive impact on the performance of the algorithms are limited, being from $N_{I_{\rm min}} = 3$ to $N_{I_{\rm max}} = 6$ for the interpolator and from $D_{\rm min} = 3$ to $D_{\rm max} = 8$ for the reduced-rank filter recursions. These values are rather insensitive to the system load (number of users), to the processing gain and work very well for all scenarios examined. In the simulations section, we will illustrate how the proposed rank adaptation algorithm performs.

Another possibility for rank adaptation is the use of the cross-validation (CV) method reported in [21]. This approach selects the filters' lengths that minimize a cost function that is estimated based on observations (training data) that have not been used in the process of building the filters themselves as described by

$$\mathcal{C}_{\rm CV}(\bar{\mathbf{w}}^{(D)}, \mathbf{v}^{(N_I)}, \mathbf{D}) = \sum_{l=1}^{i} \alpha^{i-l} \big| d(l) - \bar{\mathbf{w}}_{(i/l)}^{H,\,(D)}(l)\mathbf{D}(l)\boldsymbol{\Re}_o(l)\mathbf{v}_{(i/l)}^{*,\,(N_I)}(l)\big|^2 \quad (39)$$

We consider here the same "leave one out" approach as in [21]. For a given data record of size i, the CV approach chooses the filters $\mathbf{v}(i)$ and $\bar{\mathbf{w}}(i)$ that perform the following optimization

$$\{D_{\rm opt}, N_{I_{\rm opt}}\} = \arg \min_{\substack{n \in \{1,2,\ldots\} \\ d \in \{1,2,\ldots\}}} \mathcal{C}_{\rm CV}(\bar{\mathbf{w}}^{(D)}, \mathbf{v}^{(N_I)}, \mathbf{D}), \quad (40)$$

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

E. Automatic Selection of the Number of Branches

The goal of this subsection is to present an algorithm for automatically selecting the number of branches necessary to achieve a pre-specified performance. For each time interval i, we can select B which minimizes the instantaneous branch cost

$$\mathcal{C}_{\rm branch}(\bar{\mathbf{w}}(i), \mathbf{D}(i), \mathbf{v}(i)) = \big|e_b(i)\big|^2 \quad (41)$$

where $e_b(i) = d(i) - \bar{\mathbf{w}}^H(i)\mathbf{D}(i)\boldsymbol{\Re}_o(i)\mathbf{v}^*(i)$ is the error signal for each branch. The proposed algorithm for automatically selecting the number of branches is given by

$$\begin{aligned} B_{\rm s}(i) &= \arg \min_{1 \leq b \leq B_{\rm max}} \mathcal{C}_{\rm branch}(\bar{\mathbf{w}}(i), \mathbf{D}(i), \mathbf{v}(i)) \\ & \text{subject to } \mathcal{C}_{\rm branch}(\bar{\mathbf{w}}(i), \mathbf{v}(i), \mathbf{D}(i)) \leq \epsilon \end{aligned} \quad (42)$$

where b is an integer and $B_{\rm max}$ is the maximum number of branches allowed for the JIDF scheme, respectively, $B_{\rm s}$ is the number of branches required to attain the desired performance and ϵ is the pre-specified performance. The algorithm given in (42) determines the minimum number of branches necessary to achieve a pre-determined performance ϵ according to the cost function defined in (41). It iteratively increases the number of branches by one until the pre-determined performance ϵ is met. The parameter ϵ can be chosen as a function of the MMSE with a penalty allowed by the designer. An important measure that arises from this algorithm is the average number of branches $B_{\rm avg} = 1/Q \sum_{i=1}^{Q} B_s(i)$ with Q being the data record, which illustrates the savings in computations of the branches.

V. Analysis of The Proposed Method and Convergence Issues

In this section, we conduct an analysis of the proposed method and its convergence issues. Specifically, we study the existence of solutions and the convergence properties of the proposed scheme.

The method leads to an optimization problem with multiple solutions. Therefore, the convergence of the algorithms is not guaranteed to the global minimum since local minima may be encountered by the proposed LMS and RLS algorithms. It should be mentioned, however, that the proposed algorithms were extensively tested for many applications. It was verified in these experiments that the algorithms always converge to approximately the same filter values irrespective of the initialization. This suggests that the problem may have multiple global minima or that every point of minimum is a point of global minimum.

Another key feature of the proposed method is that it employs a combination of discrete and continuous optimization techniques, which make its convergence study extremely difficult. Even though the necessary conditions for the optimization algorithms are met [29], [30] and the cost functions used for deriving LMS and RLS algorithms are continuously differentiable, the discrete nature of the decimation and the patterns used make its theoretical analysis highly challenging. This proof is beyond the scope of this paper remains an interesting open problem.

A. Existence of Solutions

Here, we focus on the existence of the solutions to the proposed optimization problem and examine the characteristics of the critical points. For notation simplicity, we remove the index i from the filters and decimation matrix. In order to study the existence of solutions we consider the associated SES expressions in (17) and (18). We note that points of global minimum of

$$\mathcal{J}_{LS}^{(\mathbf{v},\mathbf{D},\bar{\mathbf{w}})} = \sum_{l=1}^{i} \alpha^{i-l} |d(l) - \mathbf{v}^H \boldsymbol{\Re}_o^T(l) \mathbf{D}^T \bar{\mathbf{w}}^*|^2$$

can be obtained by

$$\mathbf{v}_{opt} = \arg \min_{\mathbf{v}} \ \mathcal{J}_{LS}(\mathbf{v}, \mathbf{D}, \boldsymbol{\gamma}(\mathbf{v}, \mathbf{D}))$$

where $\mathcal{J}_{LS}(\mathbf{v}, \mathbf{D}, \boldsymbol{\gamma}(\mathbf{v}, \mathbf{D})) = \mathcal{J}_{LS}(\mathbf{v}) = \varepsilon_d - \bar{\mathbf{p}}^H(i)\bar{\mathbf{R}}^{-1}(i)\bar{\mathbf{p}}(i)$ and

$$\mathbf{w}_{opt} = \boldsymbol{\gamma}(\mathbf{v}_{opt}, \mathbf{D})$$

or

$$\mathbf{w}_{opt} = \arg\min_{\bar{\mathbf{w}}} \mathcal{J}_{LS}(\boldsymbol{\beta}(\bar{\mathbf{w}}, \mathbf{D}), \mathbf{D}, \bar{\mathbf{w}})$$

and

$$\mathbf{v}_{opt} = \boldsymbol{\beta}(\bar{\mathbf{w}}_{opt}, \mathbf{D})$$

At a minimum point the cost function $\mathcal{J}_{LS}(\mathbf{v}, \mathbf{D}, \boldsymbol{\gamma}(\mathbf{v}, \mathbf{D}))$ equals $\mathcal{J}_{LS}(\boldsymbol{\beta}(\bar{\mathbf{w}}, \mathbf{D}), \mathbf{D}, \bar{\mathbf{w}})$ and the minimum SES for the proposed structure is achieved. We further note that since $\mathcal{J}_{LS}(\mathbf{v}) = \mathcal{J}_{LS}(t\mathbf{v})$, for every $t \neq 0$, then if $\mathbf{v}^\star$ is a point of global minimum of $\mathcal{J}_{LS}(\mathbf{v})$ then $t\mathbf{v}^\star$ is also a point of global minimum. Therefore, points of global minimum (optimum interpolator filters) can be obtained by $\mathbf{v}^\star = \arg\min_{||\mathbf{v}||=1} \mathcal{J}_{LS}(\mathbf{v})$. Since the existence of at least one point of global minimum of $\mathcal{J}_{LS}(\mathbf{v})$ for $||\mathbf{v}|| = 1$ is guaranteed by the theorem of Weierstrass [29], [30], then the existence of (infinite) points of global minimum is also guaranteed for the cost function in (12). This establishes the existence of the solution of the optimization problem. Because at a minimum point (17) equals (18), the designer can consider only one of the parameter vectors, either $\bar{\mathbf{w}}$ or $\mathbf{v}$, for analysis purposes.

B. Convergence Properties

Let us now study the convergence properties of the proposed scheme and least-squares (LS) design. With respect to global convergence, a sufficient but not necessary condition is the convexity of the cost function in (12), which is verified if its Hessian matrix is positive semi-definite, that is $\mathbf{a}^H\mathbf{H}\mathbf{a} \geq 0$, for any vector $\mathbf{a}$. To illustrate the properties of the proposed method, we firstly consider the minimization of (12) with fixed interpolators and decimators. Such optimization leads to the following Hessian

$$\begin{aligned}\mathbf{H} = \frac{\partial}{\partial \bar{\mathbf{w}}^H}\frac{(\mathcal{J}_{LS}(.))}{\partial \bar{\mathbf{w}}} &= \sum_{l=1}^{i} \alpha^{i-l}\mathbf{S}_D(l)\mathbf{r}(l)\mathbf{r}^H(l)\mathbf{S}_D^H(l) \\ &= \sum_{l=1}^{i} \alpha^{i-l}\mathbf{D}(l)\boldsymbol{\Re}_o(l)\mathbf{v}^*\mathbf{v}^T\boldsymbol{\Re}_o^H(l)\mathbf{D}^H(l) \\ &= \sum_{l=1}^{i} \alpha^{i-l}\bar{\mathbf{r}}(l)\bar{\mathbf{r}}^H(l) = \bar{\mathbf{R}}(i),\end{aligned} \tag{43}$$

which is positive semi-definite and ensures the convexity of the cost function for the case of fixed interpolators. Note that $\mathbf{D}$ does not affect the convexity. Consider now the joint optimization of the interpolator $\mathbf{v}$ and receiver $\bar{\mathbf{w}}$ through an equivalent cost function to (12):

$$\tilde{\mathcal{J}}_{LS}(\mathbf{z}) = \sum_{l=1}^{i} \alpha^{i-l}|b(l) - \mathbf{z}^H\mathbf{B}(l)\mathbf{z}|^2] \tag{44}$$

where $\mathbf{B}(l) = \begin{bmatrix} \mathbf{0} & \mathbf{0} \\ \mathbf{D}(l)\boldsymbol{\Re}_o(l) & \mathbf{0}\end{bmatrix}$ is an $(N_I + D) \times (N_I + D)$ matrix and contains the contribution of the decimator $\mathbf{D}_b(i)$. The Hessian $(\mathbf{H})$ with respect to $\mathbf{z} = [\mathbf{w}^T\ \mathbf{v}^T]^T$ is

$$\begin{aligned}\mathbf{H} = \frac{\partial}{\partial \mathbf{z}^H}\frac{\partial(\tilde{\mathcal{J}}_{LS}(.))}{\partial \mathbf{z}} &= \Big(\sum_{l=1}^{i} \alpha^{i-l}(\mathbf{z}^H\mathbf{B}(l)\mathbf{z} - b(l))\mathbf{B}^H(l)\Big) \\ &+ \Big(\sum_{l=1}^{i} \alpha^{i-l}(\mathbf{z}^H\mathbf{B}^H(l)\mathbf{z} - b^*(l))\mathbf{B}(l)\Big) \\ &+ \Big(\sum_{l=1}^{i} \alpha^{i-l}\mathbf{B}(l)\mathbf{z}\mathbf{z}^H\mathbf{B}^H(l)\Big) \\ &+ \Big(\sum_{l=1}^{i} \alpha^{i-l}\mathbf{B}^H(l)\mathbf{z}\mathbf{z}^H\mathbf{B}(l)\Big).\end{aligned} \tag{45}$$

By examining the Hessian matrix $\mathbf{H}$ we note that the third and fourth terms yield positive semi-definite matrices given by

$$\mathbf{a}^H\Big(\sum_{l=1}^{i} \alpha^{i-l}\mathbf{B}(l)\mathbf{z}\mathbf{z}^H\mathbf{B}^H(l)\Big)\mathbf{a} \geq 0$$

and

$$\mathbf{a}^H\Big(\sum_{l=1}^{i} \alpha^{i-l}\mathbf{B}^H(l)\mathbf{z}\mathbf{z}^H\mathbf{B}(l)]\mathbf{a}\Big) \geq 0$$

with $\mathbf{z} \neq \mathbf{0}$, whereas the first and second terms are indefinite matrices. Thus, the cost function cannot be classified as convex. However, for a gradient search or Newton-type algorithm, a desirable property of the cost function is that it shows no points of local minimum, i.e., every point of minimum is a point of global minimum (convexity is a sufficient, but not necessary, condition for this property to hold). In order to verify that, we carried out studies that indicate that there is no local minima. Firstly, for a given set of parameters and conditions the algorithms always converge to the same minimum value. Secondly, if we consider the cost function in (12) with scalar and real parameters, we get

$$\mathcal{J}_{LS}(w, D, v) = (b - w\,D\,r\,v)^2 = b^2 - 2b\,w\,D\,r\,v + (w\,D\,r\,v)^2$$

where r is a constant. By choosing v (the "scalar" interpolator) fixed and D equal to 1, it is evident that the resulting function

$$\mathcal{J}_{LS}(w, D = 1, v) = (b - w\,c)^2$$

where c is a constant is a convex one. In contrast to that, for a time-varying scalar interpolator v the curves shown in Fig. 3 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 multi-dimensional surface.

VI. Simulations

In this section, we evaluate the proposed JIDF method and algorithms in terms of signal-to-interference-plus-noise ratio (SINR) in an application to DS-CDMA systems. We also assess the bit error rate (BER) of the JIDF scheme and algorithms and compare them with the full-rank [3], the MWF [15], [13], the AVF with non-orthogonal auxiliary vectors [20] and the MMSE, that assumes the knowledge of the channels and the noise variance.

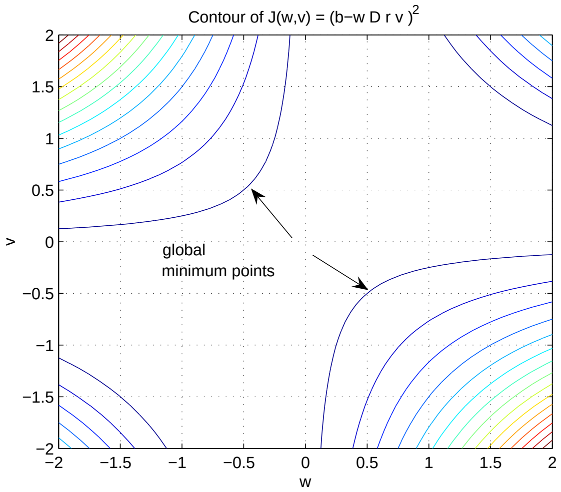

Fig. 3. Contour plots showing that the function $J_{LS}(w, D, v)$.

A. DS-CDMA System and Linear Reduced-Rank Receivers

Let us consider the uplink of a symbol synchronous DS-CDMA system with K users, QPSK modulation, N chips per symbol and L_p propagation paths [3]. We assume that the delay is a multiple of the chip rate, the channel is constant during each symbol interval and the spreading codes are repeated from symbol to symbol. The received signal after filtering by a chip-pulse matched filter and sampled at chip rate yields the M-dimensional received vector

$$\mathbf{r}(i) = \sum_{k=1}^{K} A_k b_k(i)\mathbf{C}_k \mathbf{h}_k(i) + \boldsymbol{\eta}(i) + \mathbf{n}(i), \quad (46)$$

where $M = N + L_p - 1$, $\mathbf{n}(i) = [n_1(i) \ \dots \ n_M(i)]^T$ is the complex Gaussian noise vector with zero mean and covariance matrix $E[\mathbf{n}(i)\mathbf{n}^H(i)] = \sigma^2\mathbf{I}$. The user k symbol is $b_k(i)$ and is assumed to be drawn from a general constellation. The amplitude of user k is $A_k(i)$ and $\boldsymbol{\eta}(i)$ is the intersymbol interference (ISI) for user k. The $M \times L_p$ convolution matrix $\mathbf{C}_k$ that contains one-chip shifted versions of the signature sequence for user k expressed by $\mathbf{s}_k = [a_k(1) \dots a_k(N)]^T$ and the $L_p \times 1$ vector $\mathbf{h}(i)$ with the multipath components are described by:

$$\mathbf{C}_k = \begin{bmatrix} a_k(1) & & \mathbf{0} \\ \vdots & \ddots & a_k(1) \\ a_k(N) & & \vdots \\ \mathbf{0} & \ddots & a_k(N) \end{bmatrix}, \mathbf{h}_k(i) = \begin{bmatrix} h_{k,0}(i) \\ \vdots \\ h_{k,L_p-1}(i) \end{bmatrix}. \quad (47)$$

In this model, the ISI span and contribution $\boldsymbol{\eta}(i)$ are functions of the processing gain N and L_p. We consider an uplink scenario and synchronous signals for simplicity. The system employs random sequences of length $N = 32$ and 64. All the channels are modelled as time-varying multipath channel according to Clarke's model [32] that is parameterized by the normalized Doppler frequency $f_D T$, where f_D is the Doppler frequency and T is inverse of the symbol rate. We assume $L_p = 9$ as an upper bound, which means $\mathbf{r}(i)$ has $M = N + L_p - 1 = 40$ and $M = 72$ taps, respectively. In this case, the ISI corresponds to 3 symbols namely, the current, previous and successive symbols.

The reduced-rank linear receiver design corresponds to determining an FIR filter $\bar{\mathbf{w}}(i) = \big[\bar{w}_0(i) \ \bar{w}_1(i) \ \dots \ \bar{w}_{D-1}(i)\big]^T$ with D coefficients and a $M \times D$ projection matrix $\mathbf{S}_D(i)$ that provides an estimate of the desired symbol as given by

$$\begin{aligned}\hat{b}(i) &= \mathrm{sgn}\Big(\Re\Big[\bar{\mathbf{w}}^H(i)\mathbf{S}_D^H(i)\mathbf{r}(i)\Big]\Big) + \jmath\mathrm{sgn}\Big(\Im\Big[\bar{\mathbf{w}}^H(i)\mathbf{S}_D^H(i)\mathbf{r}(i)\Big]\Big) \\ &= \mathrm{sgn}\Big(\Re\Big[x(i)\Big]\Big) + \jmath\mathrm{sgn}\Big(\Im\Big[x(i)\Big]\Big),\end{aligned} \quad (48)$$

where the operators $\Re(\cdot)$ and $\Im(\cdot)$ select the real and imaginary parts, respectively, and $\mathrm{sgn}(\cdot)$ is the signum function. The quantity $x(i) = \bar{\mathbf{w}}^H(i)\mathbf{S}_D^H(i)\mathbf{r}(i)$ is the output of the reduced-rank filtering scheme and receiver for the user of interest. In what follows, we will assume that user 1 is the user of interest and we will measure several performance metrics for assessing the performance of the proposed JIDF scheme.

B. Adjustment of Filter Parameters

In most adaptive filtering schemes, it is necessary to adjust parameters such as step size and forgetting factor. In the proposed JIDF scheme, a key issue is the setting of the number of taps or the rank of the filters $\bar{\mathbf{w}}(i)$ and $\mathbf{v}(i)$ used. We have conducted experiments in order to obtain the most adequate rank for the interpolator $\mathbf{v}(i)$, with values ranging from 3 to 8 and for the reduced-rank filter $\bar{\mathbf{w}}(i)$ with values ranging from 1 to 16. It should be remarked that using values beyond that range was unnecessary since it did not increase the performance. We consider experiments where the channels have 3 paths with relative gains at 0, −3 and −9 dB, the coefficients are obtained with Clarke's model [32] and the spacing between paths is computed with a discrete random variable between 1 and 2.

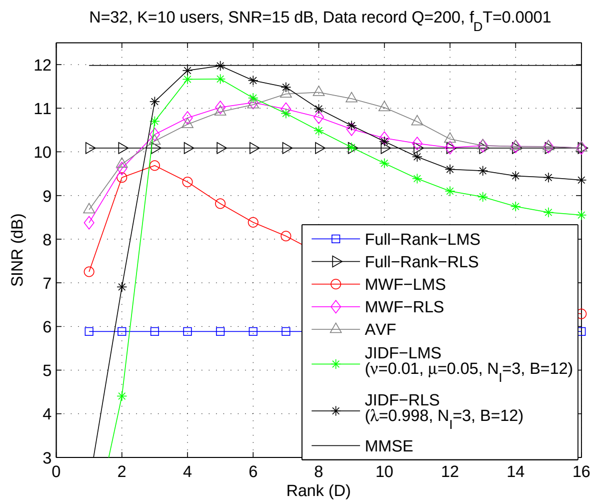

Fig. 4. SINR performance against rank (D) for the analyzed schemes using LMS and RLS algorithms.

The results in Figs. 4 and 5 for a wide range of scenarios indicate that performance is good for a small range of the number of taps in $\bar{\mathbf{w}}(i)$ and $\mathbf{v}(i)$. While the JIDF scheme is not able to construct a good subspace projection with only 1 or 2 elements in $\mathbf{v}(i)$ and $\bar{\mathbf{w}}(i)$, there is no improved modelling and the adaptation becomes slower when the size becomes reasonably large (greater than 6). Thus, for this reason and to keep the complexity low we adopt $N_I = 3$ and $D = 5$ for the next few experiments since these values yield the best performance.

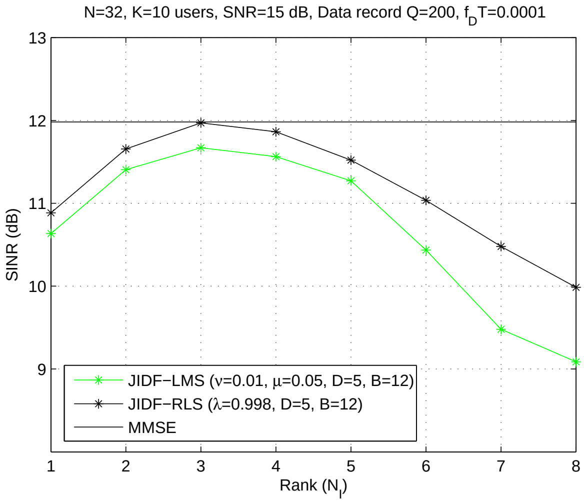

Fig. 5. SINR performance against rank (N_I) for the analyzed schemes using LMS and RLS algorithms $f_d T = 0.0001$.

C. Performance with Number of Branches and Decimation Schemes

In this part, we evaluate the performance in terms of the SINR of the proposed JIDF scheme and algorithms for different decimation schemes and the impact of the number of branches on performance. We compare the JIDF scheme and algorithms with the full-rank, the MWF [13], and the AVF [20]. The channels have 3 paths with relative gains at 0, -3 and -9 dB, the coefficients are obtained with Clarke's model [32] and the spacing between paths is computed as in the first experiments.

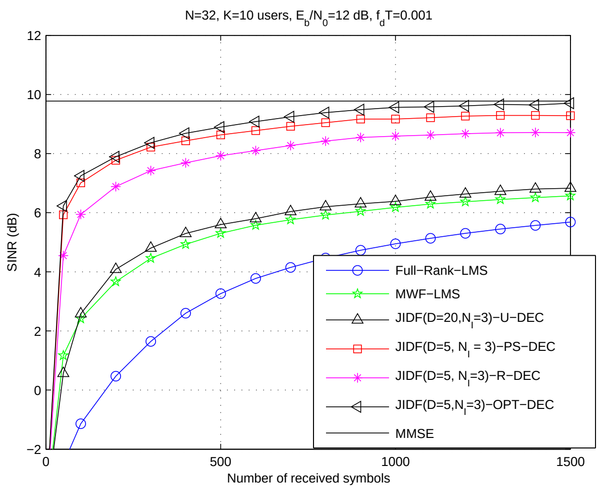

Fig. 6. SINR performance versus number of symbols with different decimation schemes for $B = 12$ and LMS algorithms.

In order to assess the proposed decimation methods, we compute the SINR performance of the JIDF scheme with the uniform (U-DEC), the random (R-DEC), the pre-stored (PS-DEC) and the optimal (OPT-DEC) decimation schemes. The results, shown in Fig. 6, indicate that the proposed scheme with the optimal decimation (OPT-DEC) achieves the best performance, followed by the proposed method with pre-stored decimation (PS-DEC), the random decimation system (R-DEC), the uniform decimation (U-DEC), the AVF, the MWF and the full-rank approach. Due to its exponential complexity, the optimal decimation algorithm is not practical and the PS-DEC is the one with the best trade-off between performance and complexity.

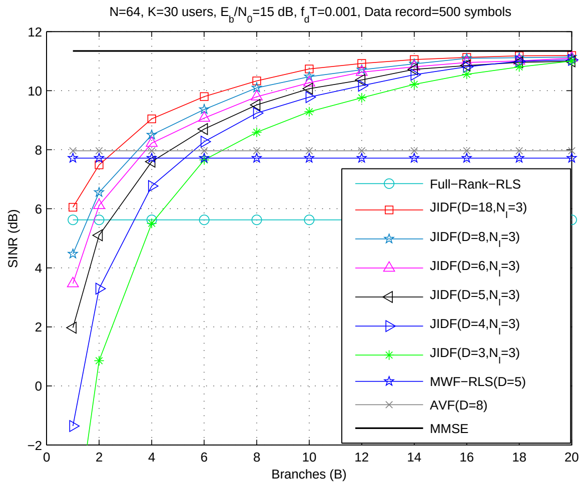

Fig. 7. SINR performance against B with RLS algorithms and PS-DEC.

In the next experiment, we evaluate the effect of the number of decimation branches B on the SINR performance of the JIDF scheme with RLS algorithms for various ranks D with a data support of 500 QPSK symbols and the PS-DEC decimation approach. The results, depicted in Fig. 7, show that the performance of the JIDF scheme is improved and approaches the optimal MMSE estimator, which assumes that the channels and the noise variance are known, as B is increased. Note that the proposed JIDF scheme and algorithms show excellent performance for a wide range of $f_d T$ values, as verified in our studies.

D. Performance with Automatic Rank and Branch Adaptation

In the next examples, illustrated in Figs. 8 and 9, we assess the performance of the proposed JIDF scheme with the automatic rank and branch adaptation algorithms described in Section IV. The channel models are the same as in the previous experiments.

The evaluation of the rank adaptation algorithms is shown in Fig. 8, where we consider the JIDF scheme with LMS algorithms, $B = 12$, $D_{\min} = 3$, $D_{\max} = 8$, and $N_{I_{\min}} = 3$ and $N_{I_{\max}} = 6$. We have one JIDF configuration using $N_I = 3$ and $D = 3$, a second configuration with $N_I = 6$ and $D = 8$, the proposed rank adaptation algorithm and the extension of the CV-based rank adaptation algorithm of [21]. The results indicate that the proposed mechanism allows the JIDF scheme to achieve fast convergence and excellent steady state performance, which

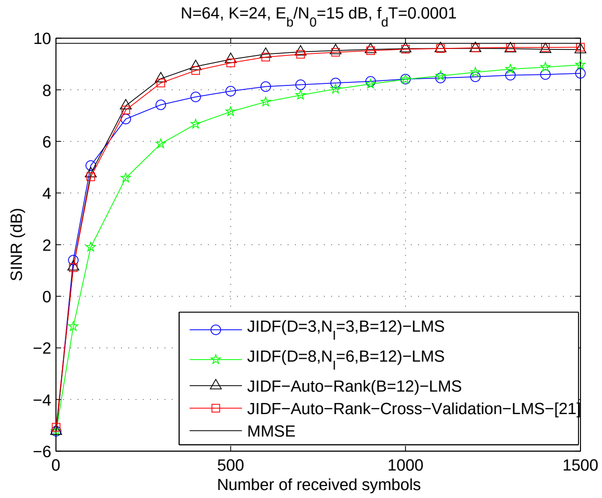

Fig. 8. SINR performance against number of symbols.

is close to the optimal MMSE. The performance of the proposed mechanism is very close to the extension of the CV-based technique of [21]. An advantage of the proposed rank adaptation algorithm over the extension of [21] is that it reduces the number of possible ranks to be used by the filters by constraining them in a pre-selected range.

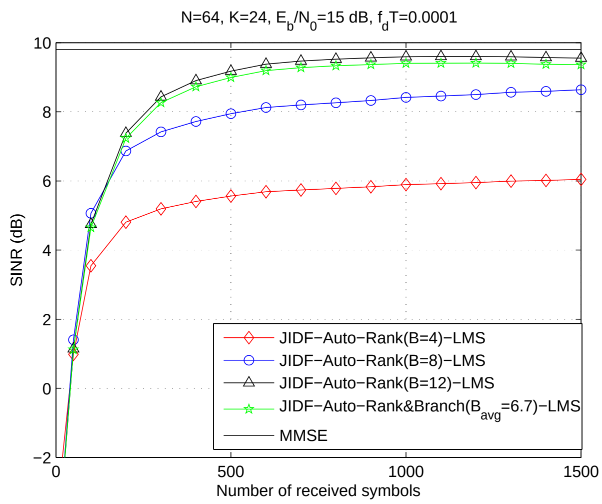

Fig. 9. SINR performance against number of symbols.

The evaluation of the branch adaptation algorithms is shown in Fig. 9 for an identical scenario to Fig. 8. We consider the JIDF scheme with LMS algorithms and the rank adaptation algorithm for different values of B, and the proposed branch adaptation algorithm. The parameter ϵ was set equal to 5% greater than the MMSE and $B_{\max} = 16$ for the experiment. The results indicate that the proposed branch adaptation mechanism allows the JIDF scheme to achieve approximately the same performance of the JIDF scheme with the rank adaptation mechanism and $B = 12$ with an average number of branches $B_{\rm avg} = 6.7$. In what follows, we will consider the automatic rank and branch adaptation mechanisms for the JIDF with the same parameters used here and the rank adaptation mechanisms proposed in [13] for the MWF and in [21] for the AVF.

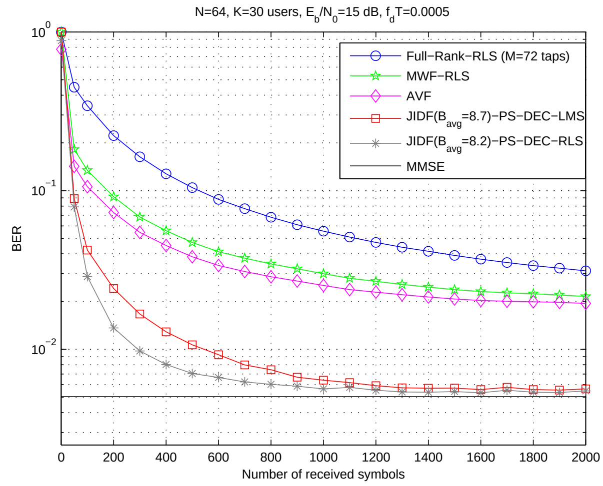

Fig. 10. BER performance against number of symbols.

The BER performance against the number of symbols is illustrated in Fig. 10. The curves show that the reduced-rank methods significantly outperform the full-rank receiver and the best performance is obtained by the proposed JIDF scheme. In particular, the JIDF with both the LMS and the RLS outperforms the remaining schemes and approaches the MMSE performance.

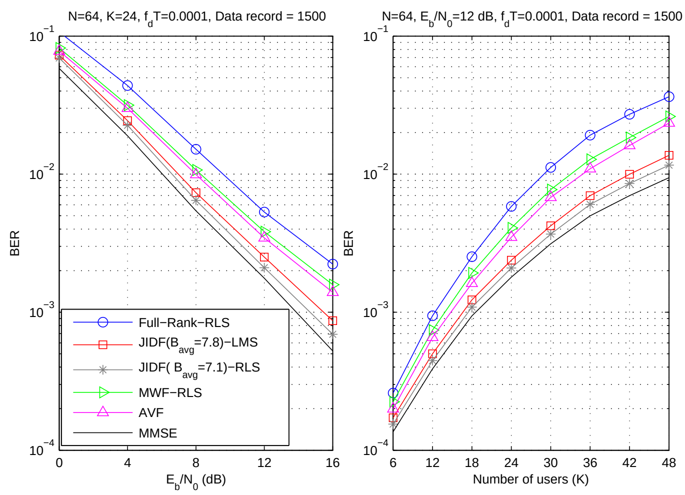

Fig. 11. BER performance versus (a) E_b/N_0 (b) number of users.

In the last experiment, we consider the BER performance against E_b/N_0 and the number of users, as depicted in Fig. 11. The BER is measured for data records of 1500 QPSK symbols and a scenario where the receivers employ pilot signals for estimating their parameters with LMS and RLS algorithms. The results show that the JIDF scheme with both LMS and RLS algorithms achieves a BER performance very close to the optimal MMSE, that assumes known channels, is followed by the AVF, the MWF-RLS and the full-rank. Specifically, the JIDF can save up to 4 dB in E_b/N_0 as compared to the AVF and the MWF-RLS for the same BER and can accommodate up to 6 more users as compared to the AVF and the MWF-RLS for the same BER.

VII. Conclusions

An adaptive reduced-rank filtering scheme was presented and applied to interference suppression in DS-CDMA systems. The proposed JIDF method is based on the reduced-rank processing of signals by jointly optimizing the interpolation, decimation and filtering tasks, and consists of a combination of continuous and discrete optimization techniques. We developed LMS and RLS algorithms along with automatic rank and branch adaptation techniques for estimating the parameters of the proposed scheme. We also provided a discussion on the existence of multiple solutions and the convergence issues of the method and algorithms. The results of simulations and the analysis indicate that the proposed reduced-rank filtering scheme allows a substantially better convergence and tracking performance than existing reduced-rank and full-rank schemes. This improvement is due to the dimensionality reduction carried out by the proposed scheme that allows the use of adaptive algorithms with very small filters. The proposed algorithms can also be applied to other applications such as MIMO systems, equalization, GPS jammer suppression and channel estimation. A proof of the convergence of the method and the algorithms remains and interesting open problem to be considered.

References

[1] S. Haykin, *Adaptive Filter Theory*, 4rd edition, Prentice-Hall, Englewood Cliffs, NJ, 2002.

[2] P. S. R. Diniz, *Adaptive Filtering: Algorithms and Practical Implementations*, 2nd ed., Kluwer, Boston, MA, 2002.

[3] 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.

[4] S. Verdu, *Multiuser Detection*, Cambridge, 1998.

[5] L. L. Scharf, "The SVD and reduced rank signal processing," Signal Processing, vol. 25, no. 2, pp. 113–133, 1991.

[6] L. L. Scharf and D. W. Tufts, "Rank reduction for modeling stationary signals," *IEEE Transactions on Acoustics, Speech and Signal Processing*, vol. ASSP-35, pp. 350-355, March 1987.

[7] L. L. Scharf and B. van Veen, "Low rank detectors for Gaussian random vectors," *IEEE Transactions on Acoustics, Speech and Signal Processing*, vol. ASSP-35, pp. 1579-1582, November 1987.

[8] A. M. Haimovich and Y. Bar-Ness, "An eigenanalysis interference canceler," *IEEE Trans. on Signal Processing*, vol. 39, pp. 76-84, Jan. 1991.

[9] X. Wang and H. V. Poor, "Blind multiuser detection: A subspace approach," *IEEE Trans. on Inf. Theory*, vol. 44, pp. 677-690, March 1998.

[10] Y. Song and S. Roy, "Blind adaptive reduced-rank detection for DS-CDMA signals in multipath channels," *IEEE Journal on Selected Areas in Communications*, vol. 17, pp. 1960-1970, November 1999.

[11] Y. Hua, M. Nikpour and P. Stoica, "Optimal reduced rank estimation and filtering," IEEE Transactions on Signal Processing, pp. 457-469, Vol. 49, No. 3, March 2001.

[12] J. S. Goldstein, I. S. Reed and L. L. Scharf, "A multistage representation of the Wiener filter based on orthogonal projections," *IEEE Transactions on Information Theory*, vol. 44, November, 1998.

[13] M. L. Honig and J. S. Goldstein, "Adaptive reduced-rank interference suppression based on the multistage Wiener filter," *IEEE Trans. on Communications*, vol. 50, no. 6, June, 2002.

[14] 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 Transactions on Signal Processing, vol. 56, no. 6, pp. 2470 - 2482 , June 2008.

[15] C. C. Hu and I. S. Reed, "Space-Time Adaptive Reduced-Rank Multistage Wiener Filtering for Asynchronous DS-CDMA," *IEEE Trans. on Sig. Proc.*, vol. 52, no. 7, July, 2004.

[16] S.-H. Wu; U. Mitra and C.-C. J Kuo, "Reduced-rank multistage receivers for DS-CDMA in frequency-selective fading channels", *IEEE Transactions on Communications*, vol. 53, no. 2, Feb. 2005.

[17] D. A. Pados and S. N. Batalama, "Low complexity blind detection of DS/CDMA signals: Auxiliary-vector receivers," *IEEE Transactions on Communications*, vol. 45, pp. 1586-1594, December 1997.

[18] D. A. Pados and S. N. Batalama, "Joint space-time auxiliary-vector filtering for DS/CDMA systems with antenna arrays," *IEEE Trans. on Communications*, vol. 47, no. 8, Sept. 1999.

[19] D. A. Pados, F. J. Lombardo and S. N. Batalama, "Auxiliary Vector Filters and Adaptive Steering for DS-CDMA Single-User Detection," *IEEE Transactions on Vehicular Technology*, vol. 48, No. 6, November 1999.

[20] D. A. Pados, G. N. Karystinos, "An iterative algorithm for the computation of the MVDR filter," *IEEE Trans. on Sig. Proc.*, vol. 49, No. 2, February, 2001.

[21] 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. on Communications*, vol. 51, no. 10, Oct. 2003, pp. 1700 - 1708.

[22] R. C. de Lamare and R. Sampaio-Neto, "Adaptive Reduced-Rank MMSE Filtering with Interpolated FIR Filters and Adaptive Interpolators", *IEEE Signal Processing Letters*, vol. 12, no. 3, March, 2005.

[23] 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. Vehicular Technology*, Vol. 56, no. 6, September 2007.

[24] R. C. de Lamare and Raimundo Sampaio-Neto, Reduced-Rank Adaptive Filtering Based on Joint Iterative Optimization of Adaptive Filters, IEEE Signal Processing Letters, Vol. 14, no. 12, December 2007.

[25] R. C. de Lamare and R. Sampaio-Neto, "Adaptive Space-Time Reduced-Rank Interference Suppression for Asynchronous DS-CDMA based on a Diversity Combined Decimation and Interpolation Scheme", *Proc. IEEE International Conference on Communications*, 2007.

[26] R. C. de Lamare and R. Sampaio-Neto, "Adaptive Reduced-Rank MMSE Parameter Estimation based on an Adaptive Diversity Combined Decimation and Interpolation Scheme," *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing*, 2007.

[27] G. H. Golub and C. F. van Loan, *Matrix Computations*, 3rd ed., The Johns Hopkins University Press, Baltimore, Md, 1996.

[28] D. S. Watkins, *Fundamentals of Matrix Computations*, 2nd ed., Wiley, 2002.

[29] D. P. Bertsekas, *Nonlinear Programming*, Athena Scientific, 2nd Ed., 1999.

[30] D. Luenberger, *Linear and Nonlinear Programming*, 2nd Ed. Addison-Wesley, Inc., Reading, Massachusetts 1984.

[31] C. T. Kelley, *Iterative Methods for Optimization* , no. 18 in Frontiers in Applied Mathematics, SIAM, Philadelphia, 1999.

[32] T. S. Rappaport, *Wireless Communications*, Prentice-Hall, Englewood Cliffs, NJ, 1996.