EURASIP Journal on Applied Signal Processing 2002:12, 1387–1400

Reduced-Rank Adaptive Filtering Using Krylov Subspace

Sergueï Burykh

*Signal and Image Processing Department of École Nationale Supérieure des Télécommunications,
Paris, 75634 Paris Cedex 13, France
Email: burykh@tsi.enst.fr*

Karim Abed-Meraim

*Signal and Image Processing Department of École Nationale Supérieure des Télécommunications,
Paris, 75634 Paris Cedex 13, France
Email: abed@tsi.enst.fr*

Received 23 January 2002 and in revised form 24 July 2002

A unified view of several recently introduced reduced-rank adaptive filters is presented. As all considered methods use Krylov subspace for rank reduction, the approach taken in this work is inspired from Krylov subspace methods for iterative solutions of linear systems. The alternative interpretation so obtained is used to study the properties of each considered technique and to relate one reduced-rank method to another as well as to algorithms used in computational linear algebra. Practical issues are discussed and low-complexity versions are also included in our study. It is believed that the insight developed in this paper can be further used to improve existing reduced-rank methods according to known results in the domain of Krylov subspace methods.

Keywords and phrases: adaptive filters, reduced-rank adaptive filters, multiuser detection, array processing, Krylov subspace methods.

1. INTRODUCTION

Adaptive filtering is widely used in signal processing applications such as array signal processing, equalization, and multiuser detection (see [1, 2, 3]). Least-square adaptive filters gained considerable attention during the last three decades and numerous algorithms were proposed [4].

The frequent problem which arises when designing an adaptive filtering system is that large observation size, and therefore, large filter length, means inevitably high computational cost, slow convergence, and poor tracking performance. However, this situation corresponds to many important practical applications such as high data rate direct-sequence code division multiple access (DS-CDMA) systems, radar or global positioning system (GPS) array processing. Reduced-rank adaptive filters provide a way out of this dilemma [3, 5]. The basic idea behind the rank reduction is to project the observation onto a lower-dimensional subspace usually defined by a set of basis vectors. The adaptation is then performed within this subspace with a low-order filter resulting in substantial computational savings, better convergence, and tracking characteristics.

This work deals with a family of closely related reduced-rank adaptive filters such as the multistage Wiener filter [6] (MSWF), the conjugate-gradient reduced-rank filter (CGRRF) [7, 8], the powers of R (POR) receiver, [9] and auxiliary-vector filters (AVF) [10, 11].

The MSWF takes its origin in a decomposition of an n-dimensional minimum mean square error (MMSE) filter into a linear combination of a full rank (n-dimensional) matched filter and a reduced-rank (($n-1$)-dimensional) MMSE filter. The latter may be further expanded into a ($n-1$)-rank matched filter, ($n-2$)-rank MMSE filter, and so on. A k-stage MSWF, which approximates the original MMSE filter, is obtained by taking the first k matched filters of the decomposition as basis vectors set. Like any other reduced-rank method, the MSWF projects the observation onto the subspace spanned by basis vectors and filters the result with a low-rank MMSE filter. AVF schemes can be viewed as iterative procedures. The whole process is initialized by the matched filter. At each step, the current filter is linearly combined with an “auxiliary” filter. Basis vectors set of an AVF is composed of the original matched filter and of the set of all auxiliary vectors. Finally, the CGRRF builds the basis set as sequence of vectors which are orthogonal in the metric defined by the received covariance matrix.

Interestingly enough, basis vectors of the MSWF, the AVF, and the CGRRF span the same subspace [12, 13]. This subspace can be shown to be Krylov subspace which is generated by taking the powers of the covariance matrix of observations on a cross-correlation (steering) vector [5]. This direct procedure of basis set generation is used in the POR receiver [9] and in the Cayley-Hamilton receiver [14] (the latter is developed in the context of centralized detection). Therefore, in the MSWF, the AVF, the CGRRF, and the POR receiver observations are projected on the same (Krylov) subspace. Our presentation, which further develops this point, is organized as follows.

In Section 2, data model is presented. Although our numerical studies use reduced-rank methods in the context of CDMA multiuser detection, the model is sufficiently general to encompass other applications, for example, radar [15] and GPS [16] space-time adaptive signal processing. Section 3 briefly introduces the reader into the problems of adaptive Wiener filtering which motivated the development of reduced-rank methods. In Section 4, the family of reduced-rank methods which are based on Krylov subspace is presented and their interrelationship is discussed. Krylov subspace methods are widely used in computational linear algebra for solving large (possibly sparse) systems of linear equations [17, 18, 19]. For that reason, the effort is made to relate presented algorithms to known algorithms of computational linear algebra. Specifically, forward recursion of the MSWF is shown to be an interpretation of the well-known Lanczos algorithm and AVF are derived from the steepest descent algorithm. In Section 5, the results of numerical studies are presented and Section 6 concludes our work.

2. DATA MODEL

Throughout the paper, the notations $*$, T, and H are used to denote the conjugate, transpose, and conjugate transpose operations, respectively.

Let $\mathbf{r}(k) = \begin{bmatrix} r_1(k) & r_2(k) & \cdots & r_N(k) \end{bmatrix}^T$ be the $N \times 1$ vector consisting of N data samples observed at time instant k, which is modeled as

$$\mathbf{r}(k) = \mathbf{H}\mathbf{s}(k) + \mathbf{n}(k), \tag{1}$$

where $\mathbf{s}(k)$ denotes the $M \times 1$ vector of source signals $s_1(k), s_2(k), \ldots, s_M(k)$, $\mathbf{H}$ is the $N \times M$ channel matrix and $\mathbf{n}(k)$ stands for the $N \times 1$ noise vector. In the sequel, $\mathbf{s}(k)$ and $\mathbf{n}(k)$ are supposed to be zero-mean and wide-sense stationary with the respective covariance matrices $E[\mathbf{s}(k)\mathbf{s}^H(k)] = \text{diag}(\epsilon_1, \epsilon_2, \ldots, \epsilon_M)$ and $E[\mathbf{n}(k)\mathbf{n}^H(k)] = \mathbf{R}_n$.

The model (1) can be used, for example, to represent M narrowband sources impinging on an N-element antenna array, or in the context of a symbol-synchronous DS-CDMA system. In the latter case, $\mathbf{s}(k)$ is the vector of signals transmitted by M system users and the ith column of channel matrix $\mathbf{H}$ represent the channel signature of user i, that is, ith spreading code convolved with ith channel impulse response. It can be easily seen that all methods discussed in this work apply (with minor modifications) to more complex models, such as an asynchronous CDMA system with strong intersymbol interference (ISI) or multirate systems.

3. MOTIVATION FOR THE REDUCED-RANK METHODS

Consider the problem of estimation of the source signal $s_1(k)$ given the observation (1). General linear estimator can be written as

$$\hat{s}_1(k) = \mathbf{w}^H\mathbf{r}(k), \tag{2}$$

where $\mathbf{w}$ is an $N \times 1$ vector (filter). The well-known full-rank Wiener filter [2] is the solution of the following linear system (normal equations):

$$\mathbf{R}\mathbf{w}_{\text{opt}}^N = \mathbf{c}, \tag{3}$$

where $\mathbf{c} = E[\mathbf{r}(k)s_1^*(k)]$ is the desired signal data cross-correlation vector and $\mathbf{R} = E[\mathbf{r}(k) \ \ \mathbf{r}^H(k)]$ is the covariance matrix of $\mathbf{r}(k)$. The important property of the Wiener filter is that it is the only filter that minimizes the mean square error (MSE) or, in other words, average error energy. In our notations, the MSE can be written as

$$J(\mathbf{w}) = E\left[\left|\left|\hat{s}_1(k) - s_1(k)\right|\right|^2\right] = \epsilon_1 + \mathbf{w}^H\mathbf{R}\mathbf{w} - \mathbf{w}^H\mathbf{c} - \mathbf{c}^H\mathbf{w}. \tag{4}$$

The Wiener filter owns its popularity not only to this property but also to its relatively simple expression as a solution of a linear system (3). However, in most practical applications such as array signal processing and CDMA multiuser detection, exact values of the covariance matrix and of the cross-correlation vector are not available. For example, in a synchronous CDMA system, such characteristics as number of CDMA users, user spreading codes, user fading, and the signal-to-noise ratio are partially or completely unknown. In radar signal processing applications, little information may be available about bearings and powers of jammer signals and so forth. Moreover, noise and signal powers as well as the channel matrix $\mathbf{H}$ may exhibit slow variations.[1] Therefore, we have to deal with some estimates of $\mathbf{R}$ and $\mathbf{c}$. By way of example, the estimate of $\mathbf{R}$ can be estimated as

$$\mathbf{R}(k) = \gamma\mathbf{R}(k-1) + (1-\gamma)\mathbf{r}(k)\mathbf{r}^H(k), \tag{5}$$

where $0 < \gamma < 1$ is the forgetting factor. As soon as exact values of $\mathbf{R}$ and $\mathbf{c}$ are replaced by the time-varying estimates $\mathbf{R}(k)$ and $\mathbf{c}(k)$, system (3) has to be resolved. Each time these estimates are updated in order to take into account the most recent samples of $\mathbf{r}(k)$. Taking into consideration high values of N, we usually encounter in practice, may be quite a problem from the computational viewpoint when using direct inversion of $\mathbf{R}(k)$. Moreover, as the system to solve has the form

$$\mathbf{R}(k)\mathbf{w}(k) = \mathbf{c}(k), \tag{6}$$

[1] Sufficiently "slow" so it is possible to keep the stationarity assumption of Section 2, at least during the observation period.

natural questions arise such as the convergence of $\mathbf{w}(k)$ to the Wiener filter $\mathbf{w}^N_{\text{opt}}$, convergence speed, and the behaviour of the solution $\mathbf{w}(k)$ in a nonstationary environment (tracking ability or "adaptivity" of the filter). These questions can only be answered while taking into account the particular method of solving (6). Unfortunately, the answers provided by conventional adaptive filtering techniques (sample matrix inversion or the recursive least square (RLS) algorithm [2]) are often unsatisfactory for applications where the amount of training data (that is, the number of observations) is limited [10], such as radar space-time adaptive processing or CDMA multiuser detection in fast fading environment.

Reduced-rank methods, as an alternative to sample matrix inversion, provide fast and efficient (approximate) solutions to (6).

4. REDUCED-RANK ADAPTIVE FILTERING USING KRYLOV SUBSPACES

In this section, we develop a new insight to recently introduced reduced-rank filtering techniques such as the MSWF, which is based on Krylov subspace methods of computational linear algebra. Our main motivation is to establish a link between these two branches of recent scientific research and to see how this relationship can be used in the context of adaptive filtering.

4.1. Filter rank reduction

Definition 1. Let $\mathcal{S}^D$ be a d-dimensional subspace of $\mathcal{C}^N$. The *reduced-rank Wiener filter in subspace* $\mathcal{S}^D$ is defined as

$$\mathbf{w}^D_{\text{opt}} \overset{\text{def}}{=} \arg\min_{\mathbf{w}\in\mathcal{S}^D} J(\mathbf{w}). \tag{7}$$

The above definition includes the full rank Wiener filter as a particular case when $\mathcal{S}^D = \mathcal{C}^N$. Let $\{\mathbf{q}_j\}$, $j = 1, \dots, D$ be an orthonormal basis of $\mathcal{S}^D$. Define $\mathbf{Q} \overset{\text{def}}{=} [\mathbf{q}_1 \ \mathbf{q}_2 \ \cdots \ \mathbf{q}_D]$. As $\mathbf{w}^D_{\text{opt}} = \mathbf{Q}\boldsymbol{\mu}$ for some $\boldsymbol{\mu} \in \mathcal{C}^D$, (7) can be rewritten as

$$\mathbf{w}^D_{\text{opt}} = \mathbf{Q}\Big(\arg\min_{\boldsymbol{\mu}\in\mathcal{C}^D} J(\mathbf{Q}\boldsymbol{\mu})\Big) = \mathbf{Q}\boldsymbol{\mu}^D_{\text{opt}}. \tag{8}$$

Substituting $\mathbf{w} = \mathbf{Q}\boldsymbol{\mu}$ into (4) yields

$$J(\mathbf{Q}\boldsymbol{\mu}) = \epsilon_1 + \boldsymbol{\mu}^H\mathbf{R}_t\boldsymbol{\mu} - \boldsymbol{\mu}^H\mathbf{c}_t - \mathbf{c}_t^H\boldsymbol{\mu}, \tag{9}$$

where the *transformed covariance matrix* $\mathbf{R}_t$ and the *transformed signal data cross-correlation vector* $\mathbf{c}_t$ are defined as

$$\begin{aligned}\mathbf{R}_t &\overset{\text{def}}{=} \mathbf{Q}^H\mathbf{R}\mathbf{Q},\\ \mathbf{c}_t &\overset{\text{def}}{=} \mathbf{Q}^H\mathbf{c}.\end{aligned} \tag{10}$$

It then follows that $\boldsymbol{\mu}^D_{\text{opt}}$ in (8) is the solution of

$$\mathbf{R}_t\boldsymbol{\mu}^D_{\text{opt}} = \mathbf{c}_t. \tag{11}$$

Therefore, the reduced-rank Wiener filter is found by solving (11) and substituting $\boldsymbol{\mu}^D_{\text{opt}}$ into (8).

Contrary to (3), (11) is a system of D linear equations. Therefore, confining the filtering operation to a low-dimensional subspace $\mathcal{S}^D$ leads to substantial gains in complexity when $D \ll N$. Better convergence and tracking properties can also be expected [10, 20]. For example, with rank reduction, we only need $2D$ observations (compared to $2N$ for the full rank filter) to be within 3 dB from the maximum filter-output signal-to-interference-plus-noise ratio (SINR) [21].

On the other hand, confining the Wiener filter to a low-dimensional subspace implies the loss of degrees of freedom of the filter and, therefore, this operation should increase the MMSE achieved by a reduced-rank method

$$J\big(\mathbf{w}^D_{\text{opt}}\big) \geq J\big(\mathbf{w}^N_{\text{opt}}\big). \tag{12}$$

As for the complexity, the computational overhead due to eventual estimation of $\mathbf{Q}$ also has to be taken into account. Therefore, "good" choice of $\mathcal{S}^D$ (and of the rank-reduction method) is always a compromise dictated by the requirements of a given application.

4.2. Reduced-rank filtering using Krylov subspace

Definition 2. Given a square matrix $\mathbf{A}$ and a nonzero vector $\mathbf{v}$, the subspace defined by

$$\mathcal{K}^D \equiv \text{span}\,\{\mathbf{v}, \mathbf{A}\mathbf{v}, \mathbf{A}^2\mathbf{v}, \dots, \mathbf{A}^{D-1}\mathbf{v}\} \tag{13}$$

is referred to as a Dth Krylov subspace associated with the pair $(\mathbf{A}, \mathbf{v})$ and is denoted $\mathcal{K}^D(\mathbf{A}, \mathbf{v})$ [17].

This work deals with a family of reduced-rank methods for which $\mathcal{S}^D = \mathcal{K}^D(\mathbf{R}, \mathbf{c})$. The natural question is: what kind of reasoning leads to this particular choice for $\mathcal{S}^D$? To answer this question, consider the gradient of the MSE (4)

$$\nabla J(\mathbf{w}) = 2(\mathbf{R}\mathbf{w} - \mathbf{c}). \tag{14}$$

Now take an arbitrary i-dimensional subspace $\mathcal{S}^i$. Let $\mathbf{w}^i_{\text{opt}}$ be the reduced-rank Wiener filter in $\mathcal{S}^i$, that is,

$$\mathbf{w}^i_{\text{opt}} = \arg\min_{\mathbf{w}\in\mathcal{S}^i} J(\mathbf{w}). \tag{15}$$

Suppose that we seek to extend the subspace $\mathcal{S}^i$ to an $(i+1)$-dimensional subspace $\mathcal{S}^{i+1}$. Since $J(\mathbf{w})$ decreases most rapidly in the direction of $-\nabla J(\mathbf{w})$, a reasonable strategy is to require that

$$\nabla J(\mathbf{w}^i_{\text{opt}}) \in \mathcal{S}^{i+1}. \tag{16}$$

It follows from (14) that, for the condition above to be satisfied, it is sufficient for $\mathcal{S}^{i+1}$ to contain the pair $(\mathbf{c}, \mathbf{R}\mathbf{w}^i_{\text{opt}})$.

Now let $\{\mathcal{S}^i,\ i = 1, 2, \dots, D\}$ be a chain of Krylov subspaces, that is, $\mathcal{S}^i = \mathcal{K}^i(\mathbf{R}, \mathbf{c})$, $i = 1, 2, \dots, D$. It is then easy to prove by induction that in this case, condition (16) is satisfied for each i within the range $1, \dots, D$. Therefore, the Krylov subspace $\mathcal{K}^D(\mathbf{R}, \mathbf{c})$ results from D steps of a sequential procedure, which (i) is initialized with the matched filter ($\mathcal{S}^1 = \mathbf{c}$);

(ii) at step i, solves the reduced-rank minimization problem (15) and extends the minimization subspace $\mathcal{S}^i$ with the gradient of the cost function (MSE) taken at the point $\mathbf{w}^i_{\text{opt}}$.

Remark 1. In many signal processing applications, the observation noise is modeled as being white. It can be shown that in this case the full-rank Wiener filter $\mathbf{w}^N_{\text{opt}}$ lies in $\mathcal{K}^M(\mathbf{R},\mathbf{c})$ where M is the number of source signals. However, M can still be high (e.g., $M = N/2$ or even exceeds N in heavily or over loaded CDMA systems) and therefore, it defines only an upper bound for D.

Remark 2. Other approaches leading to Krylov subspaces can be found in literature. For example, consider the polynomial decomposition of $\mathbf{R}^{-1}$:

$$\mathbf{R}^{-1} = \alpha_0\mathbf{I} + \alpha_1\mathbf{R} + \cdots + \alpha_{N-1}\mathbf{R}^N. \tag{17}$$

A reduced-rank filter is obtained by truncating the right-hand side of (17) to D terms and by multiplying the result by $\mathbf{c}$,

$$\mathbf{w}^N_{\text{opt}} = \mathbf{R}^{-1}\mathbf{c} \Longrightarrow \mathbf{w}^D_{\text{opt}} = \alpha'_0\mathbf{c} + \alpha'_1\mathbf{R}\mathbf{c} + \cdots \alpha'_{D-1}\mathbf{R}^{D-1}\mathbf{c}. \tag{18}$$

The coefficients $\{\alpha'_i\}$ are chosen in order to minimize the MSE (the Cayley-Hamilton Receiver of [14]) or to maximize the signal-to-interference ratio [22].

In [6], the MSWF is developed through the decomposition of the full rank Wiener filter into a linear combination of the matched filter $\mathbf{c}^N$ and of the reduced-rank Wiener filter $\mathbf{v}^{N-1}_{\text{opt}}$ in the orthogonal to $\mathbf{c}^N$ subspace

$$\mathbf{w}^N_{\text{opt}} = \beta_1\mathbf{c}^N + \beta_2\mathbf{v}^{N-1}_{\text{opt}}. \tag{19}$$

The filter $\mathbf{v}^{N-1}_{\text{opt}}$ can be further represented as a linear combination of the matched filter $\mathbf{c}^{N-1}$ (in the subspace orthogonal to $\mathbf{c}^N$) and of the Wiener filter $\mathbf{v}^{N-2}_{\text{opt}}$ of rank $N-2$ (in the subspace orthogonal to span$\{\mathbf{c}^N,\mathbf{c}^{N-1}\}$), and so on.[2] The vectors $\mathbf{c}^i$ so obtained again generate the Krylov subspace.

4.3. Exact algorithms

In this section, three reduced-rank methods are introduced. The common feature of all presented techniques is that they perform exact MSE minimization in the Krylov subspace $\mathcal{K}^D(\mathbf{R},\mathbf{c})$. Hence, these exact methods are mathematically equivalent and result in the same performance. Practical issues such as complexity, structural flexibility, and the robustness to rounding errors become of major importance when preferring one exact method to another (see the discussion in Section 4.4).

4.3.1 The powers of R (POR) receiver

The POR receiver, proposed by Honig and Xiao in [9], can be considered as the simplest Krylov subspace method of rank reduction. Vectors $\mathbf{t}_i$, $i = 1, 2, \ldots, D$, which generate the Krylov subspace $\mathcal{K}^D(\mathbf{R},\mathbf{c})$, are computed as[3]

$$\mathbf{t}_i = \mathbf{R}^{i-1}\mathbf{c}. \tag{20}$$

The algorithm is summarized in Algorithm 1. It is noteworthy that for the POR receiver, $[i, j]$th element of $\mathbf{R}_t$ can be written as

$$\mathbf{R}_t[i,j] = \mathbf{c}^H\mathbf{R}^{i+j-1}\mathbf{c}, \tag{21}$$

therefore, $\mathbf{R}_t$ is a Hankel matrix.

```
$\mathbf{t}_1 = \mathbf{c}$
$\mathbf{t}_i = \mathbf{R}\mathbf{t}_{i-1}, i = 2, \ldots, D$
$\mathbf{T} = [\mathbf{t}_1 \ \mathbf{t}_2 \ \cdots \ \mathbf{t}_D]$
$\mathbf{R}_t = \mathbf{T}^H\mathbf{R}\mathbf{T}$
$\mathbf{c}_t = \mathbf{T}^H\mathbf{c}$
Solve $\mathbf{R}_t\boldsymbol{\mu} = \mathbf{c}_t$ for $\boldsymbol{\mu}$
$\mathbf{w}^D_{\text{opt}} = \mathbf{T}\boldsymbol{\mu}$
```

Algorithm 1: Summary of the POR algorithm.

4.3.2 The multistage Wiener filter (MSWF)

The MSWF [6], shown in Figure 1 for rank $D = 4$, consists of two distinct iterative procedures. The first one (*forward recursion*) builds an orthonormal basis of the Krylov subspace $\mathcal{K}^D(\mathbf{R},\mathbf{c})$ giving the projection matrix $\mathbf{Q} = [\mathbf{q}_1 \ \mathbf{q}_2 \ \ldots \ \mathbf{q}_D]$. The second procedure (*backward recursion*) solves system (11) giving the transformed Wiener filter $\boldsymbol{\mu}^D_{\text{opt}}$ or, equivalently, the weighting of basis vectors.

Forward recursion

An orthonormal basis of $\mathcal{K}^D(\mathbf{R},\mathbf{c})$ can be constructed, for example, by applying the general Gram-Schmidt procedure to the basis (20). However, for this particular basis and for the Hermitian-symmetric $\mathbf{R}$, the Gram-Schmidt orthonormalization can be simplified. The Lanczos algorithm [17] (see Algorithm 2) represents an efficient way to compute the orthonormal basis $\{\mathbf{q}_i\}$ of the Krylov subspace $\mathcal{K}^D(\mathbf{R},\mathbf{c})$.

Forward recursion results from the following interpretation of the Lanczos algorithm. Define

$$\begin{aligned}
\mathbf{r}_0(k) &\overset{\text{def}}{=} \mathbf{r}(k),\\
\mathbf{r}_i(k) &\overset{\text{def}}{=} \left(\mathbf{I} - \sum_{j=1}^{i}\mathbf{q}_j\mathbf{q}_j^H\right)\mathbf{r}(k)\\
&= \left(\mathbf{I} - \mathbf{q}_i\mathbf{q}_i^H\right)\mathbf{r}_{i-1}(k), \quad i = 1,\ldots,D,\\
d_i(k) &\overset{\text{def}}{=} \mathbf{q}_i^H\mathbf{r}_{i-1}(k), \quad i = 1,\ldots,D.
\end{aligned} \tag{22}$$

[2] In this paper, basis vectors of the MSWF are derived differently through an orthonormalization procedure applied to $\{\mathbf{c},\mathbf{R}\mathbf{c},\ldots,\mathbf{R}^{D-1}\mathbf{c}\}$ (see Section 4.3.2).

[3] Here and in the sequel, brackets (k), indicating time dependence of an estimate, will be often omitted for the simplicity of notation.

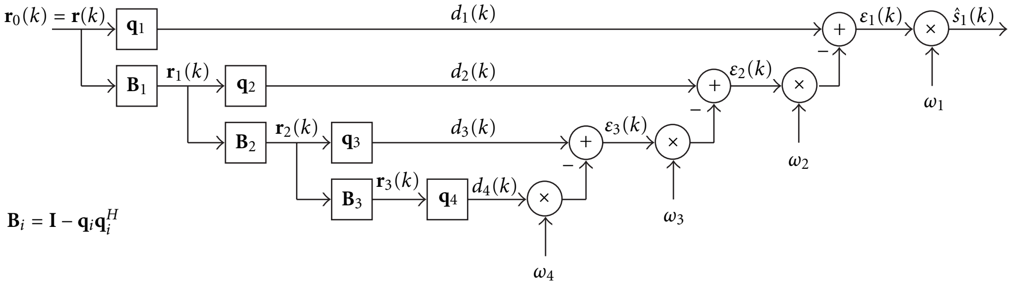

Figure 1: MSWF (rank $D = 4$).

```
Initialization:
$\mathbf{p}_1 = \mathbf{c}$
$\delta_1 = \|\mathbf{c}\|$
$\mathbf{q}_0 = \mathbf{0}$
$i := 1$
Do While ($\delta_i \neq 0$) and ($i \leq D$)
$\mathbf{q}_i = \mathbf{p}_i/\delta_i$
$\mathbf{t} = \mathbf{R}\mathbf{q}_i$
$i := i + 1$
$\mathbf{p}_i = (\mathbf{I} - \mathbf{q}_{i-1}\mathbf{q}_{i-1}^H - \mathbf{q}_{i-2}\mathbf{q}_{i-2}^H)\mathbf{t}$
$\delta_i = \|\mathbf{p}_i\|$
```

Algorithm 2: The Lanczos algorithm.

```
Initialization:
$\mathbf{p}_1 = \mathbf{c}$
$\delta_1 = \|\mathbf{c}\|$
$\mathbf{r}_0(k) = \mathbf{r}(k)$
$i := 1$
Do While ($\delta_i \neq 0$) and ($i \leq D$)
$\mathbf{q}_i = \mathbf{p}_i/\delta_i$
$\mathbf{r}_i(k) = (\mathbf{I} - \mathbf{q}_i\mathbf{q}_i^H)\mathbf{r}_{i-1}(k)$
$d_i(k) = \mathbf{q}_i^H\mathbf{r}_{i-1}(k)$
$i := i + 1$
$\mathbf{p}_i = E[\mathbf{r}_{i-1}(k)d_{i-1}^*(k)]$
$\delta_i = \|\mathbf{p}_i\|$
```

Algorithm 3: Forward recursion of the rank D MSWF.

Then $\mathbf{p}_i$ (see Algorithm 2) can be expressed as

$$\mathbf{p}_i = E\big[\mathbf{r}_{i-1}(k)d_{i-1}^*(k)\big]. \tag{23}$$

Therefore, the vector $\mathbf{p}_i$ can be viewed as the matched filter for estimating $d_{i-1}(k)$ using the reduced-rank signal $\mathbf{r}_{i-1}(k)$ [3]. Assembling together the formula for $\mathbf{q}_i$ from the Lanczos algorithm (line 1 of Algorithm 2) and (22) and (23), we obtain the forward recursion of rank D MSWF (Algorithm 3).

Backward recursion

The orthonormal basis $\{\mathbf{q}_i,\ i = 1, 2, \ldots, D\}$ [4] has a remarkable property: the transformed covariance matrix is tridiagonal [6, 18]. More precisely,

$$\mathbf{R}_t = \mathbf{Q}^H\mathbf{R}\mathbf{Q} = \begin{pmatrix} \alpha_1 & \delta_2 & & \mathbf{0} \\ \delta_2 & \alpha_2 & \ddots & \\ & \ddots & \ddots & \delta_D \\ \mathbf{0} & & \delta_D & \alpha_D \end{pmatrix}, \tag{24}$$

where

$$\begin{aligned} \delta_i &\overset{\text{def}}{=} \mathbf{q}_i^H\mathbf{R}\mathbf{q}_{i-1} = \|\mathbf{p}_i\|, \\ \alpha_i &\overset{\text{def}}{=} \mathbf{q}_i^H\mathbf{R}\mathbf{q}_i, \quad i = 1, 2, \ldots, D. \end{aligned} \tag{25}$$

Note also that the transformed signal data cross-correlation vector becomes

$$\mathbf{c}_t = \mathbf{Q}^H\mathbf{c} = \begin{bmatrix} \delta_1 & 0 & \cdots & 0 \end{bmatrix}^T. \tag{26}$$

In backward recursion of MSWF, the above-mentioned structural properties are exploited in order to solve the system

$$\mathbf{R}_t\boldsymbol{\mu}_{\text{opt}}^D = \mathbf{c}_t. \tag{27}$$

Indeed, using (24) and (26), we can write system (27) in an expanded form ($\boldsymbol{\mu}_{\text{opt}}^D = [\mu_1\ \ \mu_2\ \ \ldots\ \ \mu_D]^T$)

$$\begin{aligned} \alpha_1\mu_1 + \delta_2\mu_2 &= \delta_1, \\ \delta_{i-1}\mu_1 + \alpha_i\mu_i + \delta_{i+1}\mu_{i+1} &= 0, \quad i = 2, 3, \ldots, D-1, \\ \delta_D\mu_{D-1} + \alpha_D\mu_D &= 0. \end{aligned} \tag{28}$$

The last equation can be rewritten as

$$\mu_D = -\omega_D\mu_{D-1}, \tag{29}$$

where

$$\omega_D \overset{\text{def}}{=} \frac{\delta_D}{\alpha_D}. \tag{30}$$

[4] Basis vectors $\mathbf{q}_i$ are also called *Lanczos vectors* [18].

Substituting (29) into

$$\delta_{D-2}\mu_{D-2} + \alpha_{D-1}\mu_{D-1} + \delta_D\mu_D = 0 \tag{31}$$

yields

$$\mu_{D-1} = -\omega_{D-1}\mu_{D-2}, \tag{32}$$

where

$$\omega_{D-1} \overset{\text{def}}{=} \frac{\delta_{D-1}}{\alpha_{D-1} - \delta_D\omega_D}. \tag{33}$$

Similarly, we may show that

$$\begin{aligned} &\mu_i = -\omega_i\mu_{i-1}, \quad i = D-1, D-2, \dots, 2, \\ &\mu_1 = \omega_1, \end{aligned} \tag{34}$$

where

$$\omega_i \overset{\text{def}}{=} \frac{\delta_i}{\alpha_i - \delta_{i+1}\omega_{i+1}}, \quad i = D-1, D-2, \dots, 1. \tag{35}$$

Equations (29), (30), (34), and (35) are sufficient to solve system (27).

Another expression for ω_i can be obtained as follows. Define

$$\varepsilon_i(k) \overset{\text{def}}{=} d_i(k) - \omega_{i+1}\varepsilon_{i+1}(k), \quad i = 1, 2, \dots, D-1, \tag{36}$$

$$\varepsilon_D(k) \overset{\text{def}}{=} d_D(k). \tag{37}$$

Then

$$\begin{aligned} \omega_i &= E\left[\varepsilon_i(k)d^*_{i-1}(k)\right]/E\left[||\varepsilon_i(k)||^2\right] \\ &= \delta_i/E\left[||\varepsilon_i(k)||^2\right], \quad i = 1, 2, \dots, D, \end{aligned} \tag{38}$$

where

$$d_0(k) \overset{\text{def}}{=} s_1(k). \tag{39}$$

To show (38), it is sufficient to use definition (36) and the identities (which follow from the tridiagonality of $\mathbf{R}_t$):

$$\begin{aligned} \delta_i &= E\left[d_i(k)d^*_{i-1}(k)\right], \\ \alpha_i &= E\left[\,|d_i(k)|^2\right], \\ E\left[d_{i+j}(k)d^*_i(k)\right] &= 0, \quad |j| > 1. \end{aligned} \tag{40}$$

Assembling (36), (37), and (38) gives another form of backward recursion which is shown in Algorithm 4.

```
Initialization:
$\varepsilon_D(k) = d_D(k)$
Decrement $i = D, \dots, 1$
    $\omega_i = \delta_i/E[|\varepsilon_i(k)|^2]$
    if $i = 1$
        $\hat{s}_1(k) = \omega_1\varepsilon_1(k)$
    else
        $\varepsilon_{i-1}(k) = d_{i-1}(k) - \omega_i\varepsilon_i(k)$
```

Algorithm 4: Backward recursion of the rank D MSWF.

4.3.3 The conjugate-gradient reduced-rank filter (CGRRF)

The CGRRF [7] or conjugate-gradient implementation of the MSWF [8] are inspired directly from the conjugate-gradient algorithm (CGA) for systems of linear equations. For that reason, we start by a brief introduction into conjugate-gradient methods. For a more detailed presentation, the reader is referred to standard textbooks on computational linear algebra [18, 19].

Consider the following general iterative procedure:

$$\mathbf{w}^0 = \mathbf{0}, \tag{41}$$

$$\mathbf{w}^i = \mathbf{w}^{i-1} + c_i\mathbf{u}_i, \quad i = 1, 2, \dots, D \tag{42}$$

with the sequences of complex coefficients c_i and of vectors $\mathbf{u}_i$ chosen according to some optimization criterion.

The criterion considered here is $J(\mathbf{w}^i)$, so it is natural to require that $J(\mathbf{w}^i) \le J(\mathbf{w}^{i-1})$. Note also from (42) that $\mathbf{w}^i$ is always in $\mathcal{U}^i = \operatorname{span}\{\mathbf{u}_1, \mathbf{u}_2, \dots, \mathbf{u}_i\}$. The question is whether it is possible to choose c_i and $\mathbf{u}_i$ to give the reduced-rank Wiener filter in $\mathcal{U}^i$ or not. In other words, we require that

$$\mathbf{w}^i = \mathbf{w}^i_{\text{opt}} = \arg\min_{\mathbf{w}\in\mathcal{U}^i} J(\mathbf{w}). \tag{43}$$

The following lemma answers this question.

Lemma 1. *For the requirement* (43) *to be satisfied, it is sufficient that*

(1) $\mathbf{u}_i$ *are mutually* $\mathbf{R}$*-conjugate, that is,*

$$\mathbf{u}_i^H\mathbf{R}\mathbf{u}_j = 0, \quad i \ne j \tag{44}$$

(2) c_i *is given by*

$$c_i = \mathbf{u}_i^H\mathbf{e}_{i-1}/\mathbf{u}_i^H\mathbf{R}\mathbf{u}_i, \tag{45}$$

where

$$\mathbf{e}_i \overset{\text{def}}{=} \mathbf{c} - \mathbf{R}\mathbf{w}^i. \tag{46}$$

Proof. See [18]. □

It is easy to show that the value of the coefficient c_i as given by (45) minimizes the MSE in the direction of the line $\mathcal{L} = \{\mathbf{w}^{i-1} + c\mathbf{u}_i\}$. Therefore, condition (44) guarantees that the reduced-rank Wiener filter $\mathbf{w}^i_{\text{opt}}$ lies on $\mathcal{L}$.

Different versions of the CGA result from different ways to compute the sequence of $\mathbf{R}$-conjugate vectors $\mathbf{u}_i$ [18]. The version shown in Algorithm 5 requires only one matrix-by-vector multiplication per iteration. After D iterations of the algorithm, the sequence $\{\mathbf{w}^i_{\text{opt}}\}$ of D reduced-rank Wiener filters in $\mathcal{U}^i$ is generated.

```
Initialization:
$\mathbf{w}_{\text{opt}}^{0} = \mathbf{0}$
$\beta_1 = 0$
$\mathbf{u}_1 = \mathbf{e}_0 = \mathbf{c}/\|\mathbf{c}\|$
For $i = 1, 2, \dots, D$
    if $i > 1$
        $\beta_i = \|\mathbf{e}_{i-1}\|^2/\|\mathbf{e}_{i-2}\|^2$
        $\mathbf{u}_i = \mathbf{e}_{i-1} + \beta_i \mathbf{u}_{i-1}$
    End
$\mathbf{z}_i = \mathbf{R}\mathbf{u}_i$
$c_i = \mathbf{u}_i^H \mathbf{e}_{i-1}/\mathbf{u}_i^H \mathbf{z}_i$
$\mathbf{e}_i = \mathbf{e}_{i-1} - c_i \mathbf{z}_i$
$\mathbf{w}_{\text{opt}}^{i} = \mathbf{w}_{\text{opt}}^{i-1} + c_i \mathbf{u}_i$
```

Algorithm 5: Summary of the CGRRF.

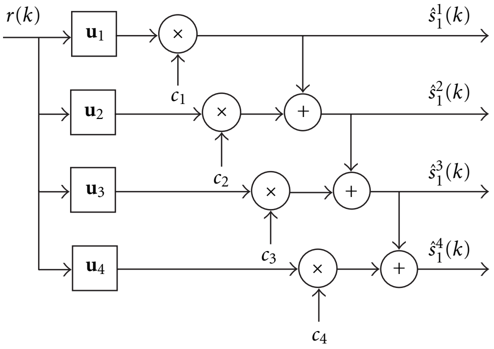

Figure 2: CGRRF (rank $D = 4$).

The following lemma establishes the equivalence between the CGA and other exact methods (MSWF, POR).

Lemma 2. *For all* $1 \leq i \leq D$, $\mathcal{U}^i = \mathcal{K}^{i-1}(\mathbf{R}, \mathbf{c})$.

Proof. See [18]. □

Therefore, the reduced-rank Wiener filter in $\mathcal{U}^i$ generated at the ith CGA iteration is also the reduced-rank Wiener filter in the Krylov subspace $\mathcal{K}^{i-1}(\mathbf{R}, \mathbf{c})$.

Basically, the CGRRF of rank D performs D CGA iterations. The CGRRF has a multistage structure, as shown in Figure 2, with the stage i computing the reduced-rank Wiener filter of the rank i and filtering the received signal to give the estimate $\hat{s}_1^i(k)$.

4.4. Discussion

The complexity of block implementations of the each exact reduced-rank method (in multiplications per block) is given in Table 1. In this table, T denotes the block size and it is assumed that the covariance matrix $\mathbf{R}$ is estimated as

$$\mathbf{R} = \frac{1}{T} \sum_{k=1}^{T} \mathbf{r}(k)\mathbf{r}^H(k). \tag{47}$$

Table 1: Complexity of exact algorithms.

Algorithm	Number of multiplications per block
SMI	$N^3/6 + TN^2$
RLS	$TN^2 + 3NT + 2T$
POR	$2TND + 3ND - N + 2D^2$
MSWF	$3TN(D-1) + (N+T)(2D-1) + D$
CGRRF	$2TND + 7ND - 3N + D$

For the reduced-rank methods, the complexity comprises the computation of basis vectors of the Krylov subspace and of the Wiener filter in the Krylov subspace. The Hankel structure of $\mathbf{R}_t$ has been accounted for in the complexity estimate for the POR receiver. The complexity of the block sample matrix inversion (SMI) method, which solves system (3) through the Cholesky decomposition of $\hat{\mathbf{R}}$, and the complexity of RLS algorithm [2] are also given for reference.

It follows from Table 1 that exact methods offer significant complexity reduction over the SMI and RLS algorithms as $D \ll N$. The analysis presented in [5, 23] shows that in practice $D = 8$ is sufficient to attain the full rank (RLS) performance for wide range of system loads (M/N) and signal-to-noise ratios (SNRs) (ϵ_i).

When some sample estimate $\mathbf{R}(k)$ is used, exact algorithms (POR, MSWF, CGRRF) provide the exact minimum of the "sample" MSE cost function

$$J(\mathbf{w}, k) = \epsilon_1 + \mathbf{w}^H \mathbf{R}(k)\mathbf{w} - \mathbf{w}^H \mathbf{c} - \mathbf{c}^H \mathbf{w} \tag{48}$$

in the "sample" Krylov subspace

$$\mathcal{K}^D(\mathbf{R}(k), \mathbf{c}) = \operatorname{span}\{\mathbf{c}, \mathbf{R}(k)\mathbf{c}, \dots, \mathbf{R}^{D-1}(k)\mathbf{c}\}. \tag{49}$$

As the minimization subspace and the cost function are common to these algorithms, they are mathematically equivalent and they ideally result in the same reduced-rank solution. However, the ways of obtaining this solution are different which may have an impact when implementing exact algorithms using finite precision arithmetic. For example, POR basis vectors $\mathbf{t}_i$, $i = 1, 2, \dots, D$ are not orthonormal. For small sample sizes, these basis vectors are nearly dependent [9] and the matrix $\mathbf{R}_t$ often becomes ill-conditioned. In [9], an adaptive rank selection technique was proposed to overcome this difficulty. Another obstacle to practical use of the POR algorithm is significant disparity in norms of $\mathbf{t}_i$. Indeed, according to (20), $\|\mathbf{t}_i\|$ grows exponentially with i thus, complicating the fixed point implementation of the algorithm. Clearly, dependence of basis vectors and norm disparity can be eliminated by some kind of orthonormalisation with additional computational cost.

Compared to the MSWF and the POR receiver, the CGRRF has the advantage of computing Wiener filters of *all* ranks ranging from 1 to D, therefore, in the CGRRF filter, outputs (symbol estimates) of different ranks $\hat{s}_1^i(k)$, $i = 1, \dots, D$ are simultaneously available, as shown in Figure 2.

This property simplifies real-time filter rank selection by measuring the SINR at the output of each stage and adapting the filter rank D to achieve the given target SINR. Moreover, the CGRRF of any rank $i < D$ is always at hand while for the MSWF and POR system (11) has to be resolved for each value of i (for the MSWF, weighting coefficients ω_i resulting from backward recursion have to be recomputed).

4.5. Auxiliary-vector filters (AVF)

Consider the system of normal equations (3). Note that the Wiener filter, which minimizes the MSE given by (4), also minimizes the function

$$J^0(\mathbf{w}) = 0.5\left(\mathbf{w}^H\mathbf{R}\mathbf{w} - \mathbf{w}^H\mathbf{c}\right). \tag{50}$$

Starting from an arbitrary filter $\mathbf{w}^0$, the *Newton-Raphson* [19] iteration gives the exact solution to (3)

$$\mathbf{w}_{\text{opt}}^N = \mathbf{w}^0 - \left(\nabla^2 J^0(\mathbf{w}^0)\right)^{-1}\nabla J^0(\mathbf{w}^0). \tag{51}$$

Substituting $\mathbf{R} = \nabla^2 J^0(\mathbf{w}^0)$ into the above expression yields

$$\mathbf{w}_{\text{opt}}^N = \mathbf{w}^0 - \mathbf{v}^1, \tag{52}$$

where $\mathbf{v}^1$ is the solution of

$$\mathbf{R}\mathbf{v}^1 = \nabla J^0(\mathbf{w}^0). \tag{53}$$

It is clear that to solve the original MMSE minimization problem, we have to provide the solution to (53). Suppose that the following approximation is used instead:

$$\mathbf{v}^1 = \mathbf{R}^{-1}\nabla J^0(\mathbf{w}^0) \approx c_1\mathbf{g}_1, \tag{54}$$

where the unit norm *auxiliary vector* $\mathbf{g}_1$ and constant c_1 are defined as

$$\mathbf{g}_1 = \nabla J^0(\mathbf{w}^0)/\|\nabla J^0(\mathbf{w}^0)\|, \tag{55}$$

$$c_1 = -\mathbf{g}_1^H\mathbf{e}_0/\mathbf{g}_1^H\mathbf{R}\mathbf{g}_1, \tag{56}$$

$$\mathbf{e}_0 = \mathbf{c} - \mathbf{R}\mathbf{w}^0. \tag{57}$$

Substituting (54) into (51) gives

$$\mathbf{w}_{\text{opt}}^N \approx \mathbf{w}^1 = \mathbf{w}^0 - c_1\mathbf{g}_1. \tag{58}$$

It can be shown that c_1 as given by (56) minimizes $J^0(\mathbf{w})$ along the line $\mathcal{L} = \{\mathbf{w}^0 + c\mathbf{g}_1\}$. The "approximate" Newton-Raphson iteration is therefore an iteration of the method of steepest descent [18, 19]. Clearly, we may continue to iterate as

$$\mathbf{w}^i = \mathbf{w}^{i-1} - c_i\mathbf{g}_i, \tag{59}$$

where

$$\mathbf{g}_i = \nabla J^0(\mathbf{w}^{i-1})/\|\nabla J^0(\mathbf{w}^{i-1})\|, \tag{60}$$

$$c_i = -\mathbf{g}_i^H\mathbf{e}_{i-1}/\mathbf{g}_i^H\mathbf{R}\mathbf{g}_i, \tag{61}$$

$$\mathbf{e}_i = \mathbf{c} - \mathbf{R}\mathbf{w}^i. \tag{62}$$

```
Initialization:
$\mathbf{w}_0 = \mathbf{c}/\|\mathbf{c}\|$
For $i = 1, 2, \ldots, D-1$
$\mathbf{t} = \mathbf{R}\mathbf{w}_{i-1} - \mathbf{c}$
$\delta = \|\mathbf{t}\|$
$\mathbf{g}_i = \mathbf{t}/\delta$
$c_i = \delta/\mathbf{g}_i^H\mathbf{R}\mathbf{g}_i$
$\mathbf{w}_i = \mathbf{w}_{i-1} - c_i\mathbf{g}_i$
```

Algorithm 6: The AVF (steepest descent) algorithm.

Substituting $\nabla J^0(\mathbf{w}) = \mathbf{R}\mathbf{w} - \mathbf{c}$ in (60) and setting $\mathbf{w}^0 = \mathbf{c}/\|\mathbf{c}\|$ leads to the algorithm given in Algorithm 6, denoted here as AVF.

It should be noted that our derivation is based on an unconstrained minimization of the MSE. In [11], the filter output energy $(\mathbf{w}^i)^H\mathbf{R}\mathbf{w}^i$ is minimized under the constraint $(\mathbf{w}^i)^H\mathbf{c} = 1$. The resulting algorithm, denoted here as the minimum variance AVF (MVAVF), computes auxiliary vectors as

$$\mathbf{g}_i = \left(\mathbf{I} - \mathbf{c}\mathbf{c}^H\right)\nabla J'(\mathbf{w}^{i-1})/\|\nabla J'(\mathbf{w}^{i-1})\|, \tag{63}$$

where $J'(\mathbf{w}) = \mathbf{w}^H\mathbf{R}\mathbf{w}$. Clearly, the term $(\mathbf{I} - \mathbf{c}\mathbf{c}^H)$ in (63) guarantees the orthogonality between $\mathbf{g}_i$ and $\mathbf{c}$ and leads to the constant response in the direction of $\mathbf{c}$,

$$\left(\mathbf{w}^i\right)^H\mathbf{c} = \left(\mathbf{w}^0\right)^H\mathbf{c} = \text{const.} \tag{64}$$

Also, we may impose the orthogonality between $\mathbf{g}_i$,

$$\mathbf{g}_i \perp \mathbf{g}_j, \quad j = 1, 2, \ldots, i-1. \tag{65}$$

These constraints lead to

$$\mathbf{g}_i = \left(\mathbf{I} - \sum_{j=1}^{i-1}\mathbf{g}_j\left(\mathbf{g}_j\right)^H\right)\nabla J^0(\mathbf{w}^{i-1})/\|\nabla J^0(\mathbf{w}^{i-1})\|. \tag{66}$$

The algorithm which employs $\mathbf{g}_i$, as defined above, is denoted here as the constrained AVF (CAVF), and its minimum variance counterpart, originally derived in [10], will be referred to as the CMVAVF (see Algorithm 7).

It is easy to see that, for all variants of the AVF, $\mathbf{w}^i \in \operatorname{span}\{\mathbf{c}, \mathbf{g}_1, \mathbf{g}_2, \ldots, \mathbf{g}_i\} = \mathcal{K}^i(\mathbf{R}, \mathbf{c})$. Moreover, for the CMVAVF, $\mathbf{g}_i = \mathbf{q}_i$, $i = 1, 2, \ldots, D$, where $\mathbf{q}_i$ (Lanczos vectors) are generated by the algorithm of Algorithm 2.Therefore, basis vectors of the CMVAVF and of the MSWF coincide. The latter fact, however, does not imply the equivalence of an AVF with D auxiliary vectors to an exact method of rank $D+1$. The main difference resides in the manner the reduced-rank filter solution is obtained. Indeed, exact methods perform the MSE minimization over the whole Krylov subspace $\operatorname{span}\{\mathbf{c}, \mathbf{g}_1, \mathbf{g}_2, \ldots, \mathbf{g}_D\}$ thus, giving the exact reduced-rank Wiener filter, while an AVF algorithm with D auxiliary vectors performs D successive line-search optimizations in $\operatorname{span}\{\mathbf{w}_{i-1}, \mathbf{g}_i\}$.

```
Initialization:
$\mathbf{w}_0 = \mathbf{c}/\|\mathbf{c}\|$
For $i = 1, 2, \ldots, D-1$
MVAVF ([11]): $\mathbf{t} = (\mathbf{I} - \mathbf{c}\mathbf{c}^H)\mathbf{R}\mathbf{w}_{i-1}$
Constrained MVAVF ([10]): $\mathbf{t} = (\mathbf{I} - \mathbf{c}\mathbf{c}^H - \sum_{j=1}^{i-1} \mathbf{g}_j\mathbf{g}_j^H)\mathbf{R}\mathbf{w}_{i-1}$
$\mathbf{g}_i = \mathbf{t}/\|\mathbf{t}\|$
$c_i = \mathbf{g}_i^H\mathbf{R}\mathbf{w}_{i-1}/\mathbf{g}_i^H\mathbf{R}\mathbf{g}_i$
$\mathbf{w}_i = \mathbf{w}_{i-1} - c_i\mathbf{g}_i$
```

Algorithm 7: Minimum variance AVF algorithms.

Table 2: Complexity of auxiliary vector algorithms.

AVF algorithm	Number of multiplications per block
AVF (steepest descent)	$4TN(D-1) + 4N(D-1) + D - 1$
MVAVF	$4TN(D-1) + 6N(D-1) + 2(D-1)$
CAVF	$4TN(D-1) + N(D-1)^2 + 4N(D-1) + 2(D-1)$
CMVAVF	$4TN(D-1) + N(D-1)^2 + 6N(D-1) + 2(D-1)$

The complexity of the four auxiliary vector algorithms (in block implementation) is given in Table 2 in multiplications per block of size T. For the algorithms compared in this table, the number of auxiliary vectors $n_{\mathrm{AV}} = D - 1$.

4.6. *Recursive algorithms*

Various recursive implementations can be derived from the algorithms introduced in the previous section. Note that each algorithm considered previously is initialized with the matched filter ($\mathbf{w}_0 = \mathbf{c}$). After D iterations, we obtain the reduced-rank Wiener filter $\mathbf{w}_1^D$ in $\mathcal{K}^D(\mathbf{R}, \mathbf{c})$. We may initialize the algorithm with $\mathbf{w}_1^D$ and run another D iterations, thus giving the reduced-rank Wiener filter $\mathbf{w}_2^D$ in $\mathcal{K}^D(\mathbf{R}, \mathbf{w}_1^D)$. Next D iterations will minimize the MSE in $\mathcal{K}^D(\mathbf{R}, \mathbf{w}_2^D)$ and so on. Another kind of recursion can be used in adaptive sample-by-sample processing when the reduced-rank method at time instant n is initialized with the filter obtained at time $(n-1)$. An example of this technique is the "CG1" variant of the CGA described in [24].

4.7. *Approximate sample-by-sample implementations*

Sample-by-sample implementation of the MSWF (adaptive residual correlation or the adaptive ResCor algorithm [25]) can be derived as follows. Recall the expressions for the matched filters $\mathbf{p}_i$ (Section 4.3.2),

$$\begin{aligned}\mathbf{p}_i &= (\mathbf{I} - \mathbf{q}_{i-1}\mathbf{q}_{i-1}^H - \mathbf{q}_{i-2}\mathbf{q}_{i-2}^H)\mathbf{R}\mathbf{q}_{i-1} \\ &= E[\mathbf{r}_{i-1}(k)d_{i-1}^*(k)], \quad i = 1, 2, \ldots, D.\end{aligned} \tag{67}$$

We may use a stochastic approximation to compute the expectation in (67), for example, as

$$\mathbf{p}_i(k) = \gamma\mathbf{p}_i(k-1) + (1-\gamma)\mathbf{r}_{i-1}(k)d_{i-1}^*(k), \tag{68}$$

where $0 < \gamma < 1$ is the forgetting factor. Note that this update does not require costly matrix vector multiplications. The same trick can be applied to the expectation which appears in backward recursion of the MSWF

$$\omega_i(k) = \delta_i(k)/E\Big[\big\|\varepsilon_i(k)\big\|^2\Big] \approx \delta_i(k)/\xi(k), \quad i = D, \ldots, 1, \tag{69}$$

```
Forward recursion
    Initialization:
    $\mathbf{p}_1(k) = \mathbf{c}(k)$
    $\delta_1(k) = \|\mathbf{c}(k)\|$
    $\mathbf{r}_0(k) = \mathbf{r}(k)$
    $i := 1$
    Do While $(\delta_i \neq 0)$ and $(i \leq D)$
    $\mathbf{q}_i(k) = \mathbf{p}_i(k)/\delta_i(k)$
    $\mathbf{r}_i(k) = (\mathbf{I} - \mathbf{q}_i(k)\mathbf{q}_i^H(k))\mathbf{r}_{i-1}(k)$
    $d_i(k) = \mathbf{q}_i^H(k)\mathbf{r}_{i-1}(k)$
    $i := i + 1$
    $\mathbf{p}_i(k) = \gamma\mathbf{p}_i(k-1) + (1-\gamma)\mathbf{r}_{i-1}(k)d_{i-1}^*(k)$
    $\delta_i(k) = \|\mathbf{p}_i(k)\|$
Backward recursion
    Initialization:
    $\varepsilon_D(k) = d_D(k)$
    Decrement $i = D, \ldots, 1$
    $\xi_i(k) = \gamma\xi_i(k-1) + (1-\gamma)\|\varepsilon_i(k)\|^2$
    $\omega_i(k) = \delta_i(k)/\xi_i(k)$
      if $i = 1$
        $\hat{s}_1(k) = \omega_1(k)\varepsilon_1(k)$
      else
        $\varepsilon_{i-1}(k) = d_{i-1}(k) - \omega_i(k)\varepsilon_i(k)$
```

Algorithm 8: Summary of the adaptive ResCor algorithm.

```
Initialization:
$\mathbf{w}_{\mathrm{opt}}^{0}(k) = \mathbf{0}$
$\beta_1(k) = 0$
$\mathbf{u}_1(k) = \mathbf{e}_0(k) = \mathbf{c}(k)/\|\mathbf{c}(k)\|$
For $i = 1, 2, \ldots, D$
    if $i > 1$
        $\beta_i(k) = \|\mathbf{e}_{i-1}(k)\|^2/\|\mathbf{e}_{i-2}(k)\|^2$
        $\mathbf{u}_i(k) = \mathbf{e}_{i-1}(k) + \beta_i(k)\mathbf{u}_{i-1}(k)$
    End
$\mathbf{z}_i(k) = \gamma\mathbf{z}_i(k-1) + (1-\gamma)\mathbf{r}(k)\mathbf{r}^H(k)\mathbf{u}_i(k)$
$\alpha_i(k) = \gamma\alpha_i(k-1) + (1-\gamma)|\mathbf{u}_i^H(k)\mathbf{r}(k)|^2$
$c_i(k) = \mathbf{u}_i^H(k)\mathbf{e}_{i-1}(k)/\alpha_i(k)$
$\mathbf{e}_i(k) = \mathbf{e}_{i-1}(k) - c_i(k)\mathbf{z}_i(k)$
$\mathbf{w}_{\mathrm{opt}}^{i}(k) = \mathbf{w}_{\mathrm{opt}}^{i-1}(k) + c_i(k)\mathbf{u}_i(k)$
```

Algorithm 9: Summary of adaptive CGRRF.

Table 3: Complexity of some sample-by-sample algorithms.

Algorithm	Number of multiplications per sample
LMS	$2N$
Adaptive CMVAVF	$8ND$
Adaptive MVAVF	$7ND$
Adaptive ResCor	$7ND$
Adaptive CGRRF	$9ND - 3N$

Table 4: Simulation parameters.

Parameter	Notation	Value
Spreading factor	S_f	16
Chip period	T_c	0.25×10^{-6} s
Carrier frequency	f_c	2×10^9 Hz

where $\xi_i(k)$ is updated as

$$\xi_i(k) = \gamma\xi_i(k-1) + (1-\gamma)\|\varepsilon_i(k)\|^2. \tag{70}$$

The resulting algorithm (A-ResCor) is summarized in Algorithm 8. Low-complexity version of the CGRRF is given in Algorithm 9. Similarly, approximate adaptive implementations of the AVF can be obtained [12, 26].

In Table 3, the complexity of sample-by-sample algorithms is given in multiplications per sample (the complexity of the LMS algorithm is given for comparison).

5. COMPUTER SIMULATIONS

The considered techniques were tested in a scenario which models joint (multiuser) detection in time division CDMA (TD-CDMA), universal mobile telecommunications system (UMTS), terrestrial radio access (UTRA), time division duplex (TDD) mode [27]. The parameters of a simulated DS-CDMA system are summarized in Table 4.

Table 5: Tap delays and relative average powers of the vehicular A channel model.

Tap	Delay (ns)	Power (dB)
1	0	0
2	310	−1
3	710	−9
4	1090	−10
5	1730	−15
6	2510	−20

Each of $M = 4$ system users transmits an i.i.d. sequence of QPSK-modulated symbols. The vehicular environment (Type A) with a mobile speed of 120 km/h was modeled [28, 29]. At this speed, the coherence time of the channel is about 500 symbols. The channel is modeled as a tapped delay line with a fixed delay. Each tap coefficient is generated using Jakes fading model [30]. The delays and relative powers of the taps are given in Table 5. The channel noise is modeled as being white.

The receiver employs two diversity antennas, that is, the signals impinging on the antennas are assumed to be uncorrelated. This is reflected in the simulation setup by modelling the path from the transmitter to each receiver antenna by an independently fading, point-to-point channel model. Let $\mathbf{r}_1(t)$ and $\mathbf{r}_2(t)$ denote the respective antenna element outputs. After the matched filtering and chip-rate sampling, the combined received signal vector $\mathbf{r}(k) = [\mathbf{r}_1^T(k) \ \ \mathbf{r}_2^T(k)]^T$ is described by (1) with the observation size $N = 2S_f = 32$. Note that due to the users' motion, the channel matrix $\mathbf{H}$ is time-varying. The channel signature $\mathbf{h}_1$ of the desired user (first column of $\mathbf{H}$ in (1)) is supposed to be known at the receiver.

Exact and SMI algorithms, compared here, use (5) to estimate the covariance matrix of the received signal with the forgetting factor $\gamma = 0.995$. This value of γ is also used in approximate adaptive implementations. Initially, $\mathbf{R}(0) = \delta\mathbf{I}$ where the diagonal loading parameter δ ensures that $\mathbf{R}(k)$ is inversible for small k.

In Figures 3 and 4, the convergence and tracking performance of the exact reduced-rank methods (ERRM, which include the POR, MSWF, and CGRRF) and of the conventional SMI technique is explored. In these figures, the bit error rate (BER) of the first user, averaged over the independent realizations of noise, transmitted symbols, fading waveforms, and random binary spreading codes is plotted versus the number of the received observations (k). The BER attained by the RAKE receiver is given for comparison. For Figure 4, the first user has the SNR of 9 dB, while the interfering users have the SNR of 21 dB, and for Figure 3, everybody is at 16 dB.

The curves of Figure 3 illustrate two distinct zones: the zone of initial convergence (first 150 symbols periods), followed by the zone of tracking. The first zone is characterized by minor accumulated changes in the propagation channel

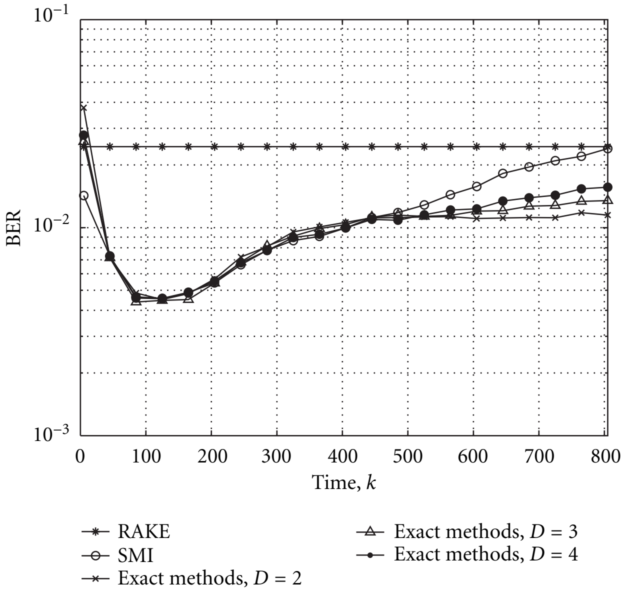

Figure 3: Convergence of ERRM and conventional methods. $\text{SNR}_1 = \text{SNR}_2 = \text{SNR}_3 = \text{SNR}_4 = 16\,\text{dB}$.

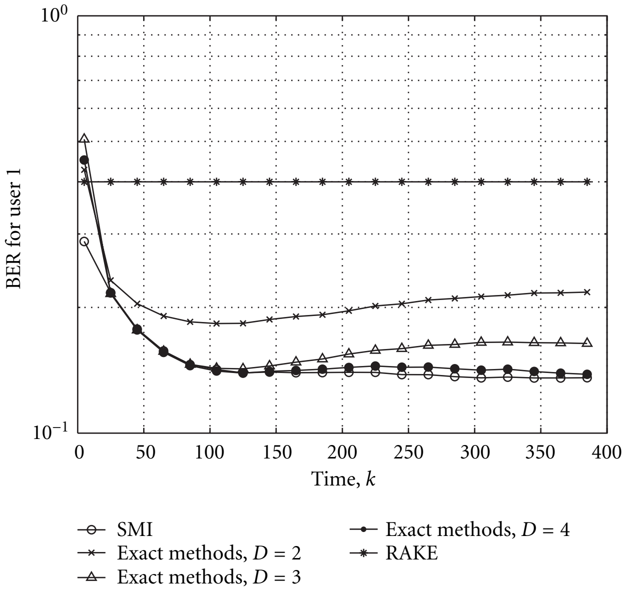

Figure 4: Convergence of ERRM and conventional methods. $\text{SNR}_1 = 9\,\text{dB}$, $\text{SNR}_2 = \text{SNR}_3 = \text{SNR}_4 = 21\,\text{dB}$.

so that the performance depends mainly on the algorithm's convergence rate. It can be seen that the reduced-rank filters converge at least as rapidly as the full rank SMI filter.

In the second zone, the channel has been significantly driven off its initial value and tracking ability becomes dominant. We see that for moderate multiple-access interference (MAI) levels (Figure 3), tracking ability is dramatically improved by the rank reduction with lower filter ranks performing the best. For strong MAI (Figure 4), accurate modelling of the interference (which is better obtained with higher ranks) prevails over the tracking performance, and the full rank solution gives the lowest BER. The latter observation also applies to the next simulation example.

In Figures 5 and 6, we compare the steady-state performance of ERRM and SMI. As a performance measure, we used the BER of the desired user over 10^6 symbol periods and averaged over random binary spreading codes. In Figure 5, this BER is plotted versus the SNR (common to all system users), while in Figure 6, only SNR of the interfering users (Interference-to-Noise Ratio, or INR) changes with the SNR of the desired user fixed at 9 dB.

It can be concluded that for low SNR (INR), ERRM of low ranks (and, in particular, the RAKE which corresponds to $D = 1$) perform better. In this case, white noise dominates the MAI and the gain achieved through multiuser detection is small compared to the loss resulting from the misadjustment noise. The situation changes for high SNR (INR) when more filter degrees of freedom are required in order to cope with the strong MAI. It follows from the presented figures that although reduced-rank filters are not near-far resistant, they can provide significant gains over the SMI and RAKE receivers over wide ranges of MAI levels.

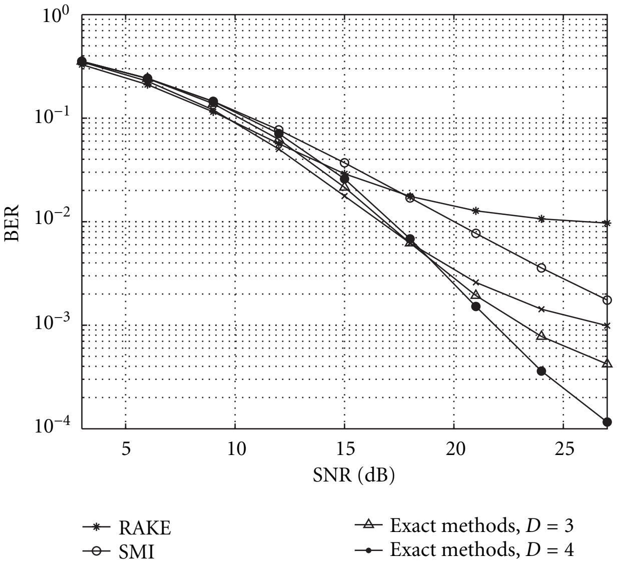

Figure 5: BER versus SNR for ERRM and conventional methods.

Our next experiment (Figure 7) compares the performance of sample-by-sample reduced-rank implementations: the adaptive ResCor (A-ResCor), the adaptive CMVAVF (A-CMVAVF), the adaptive MVAVF (A-MVAVF), and the adaptive CGRRF (A-CGRRF) of rank $D = 4$. The performance measure used in this experiment was the filter output SINR averaged over the independent realizations of noise, transmitted symbols, fading waveforms, and random binary spreading codes. The SINR at time instant k is

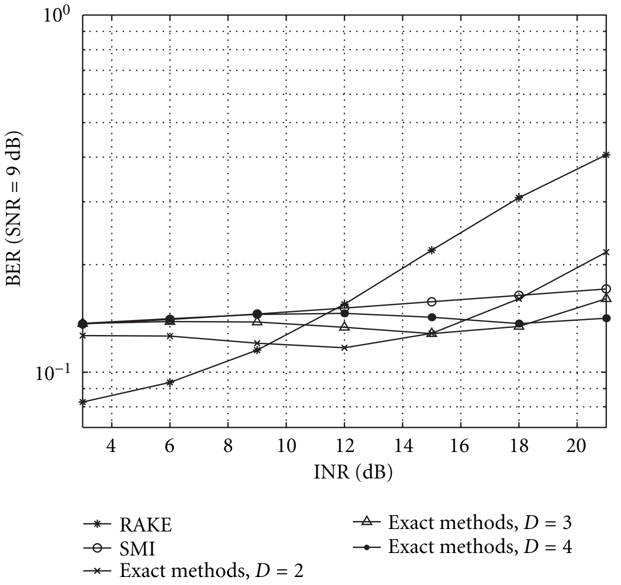

Figure 6: BER versus interference-to-noise ratio for ERRM and conventional methods. $\text{SNR}_1 = 9\,\text{dB}$.

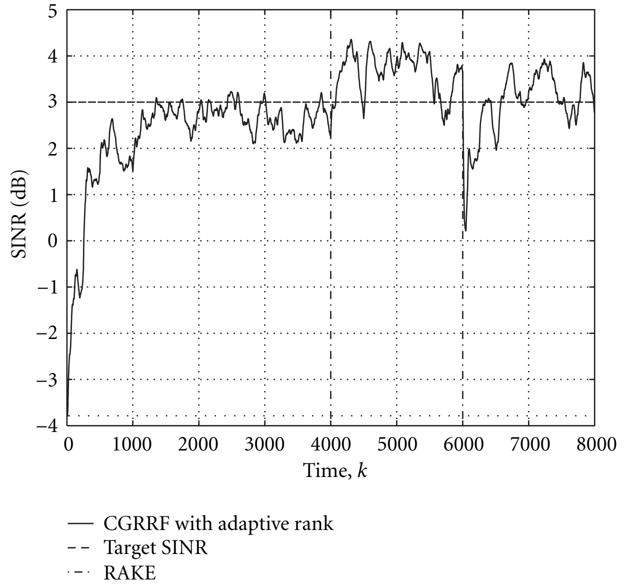

Figure 8: Averaged SINR versus time for the CGRRF with adaptive rank selection.

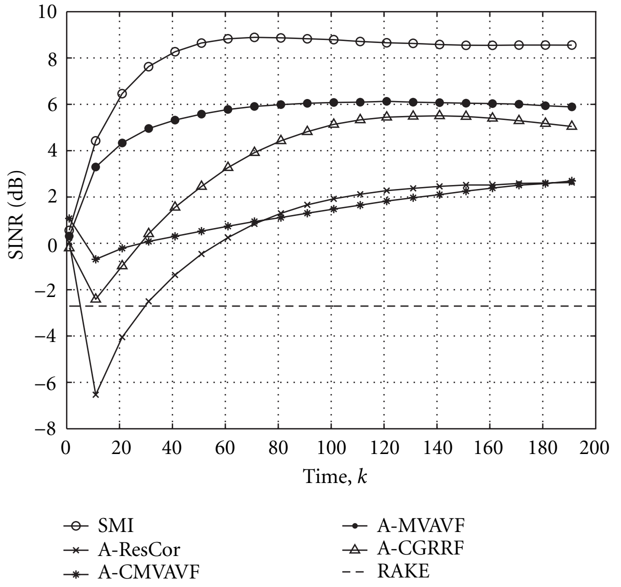

Figure 7: Output SINR versus time for sample-by-sample algorithms.

computed as

$$\text{SINR}(k) = \frac{\epsilon_1 \left| \mathbf{w}^H(k)\mathbf{h}_1(k) \right|^2}{\mathbf{w}^H(k)\mathbf{R}_{I+N}\mathbf{w}(k)}, \tag{71}$$

where $\mathbf{w}(k)$ is the filter estimate and $\mathbf{R}_{I+N}$ is the interference plus noise covariance matrix. In this experiment, the desired user is at 9 dB SNR, while the interferers are at 21 dB. The performance attained by the full rank and RAKE filters is also given for reference. Interestingly enough, for the given simulation setup algorithms which use nonorthogonal sets of basis (auxiliary) vectors (A-CGRRF, A-MVAVF) result in better performance.

Figures 5 and 6 suggest that adaptive rank selection (as a function of MAI level) can be used in order to improve the near-far resistance of reduced-rank filters. This solution is demonstrated in our next experiment, where the rank of CGRRF is adapted in order to maintain constant filter output SINR.

Specifically, each 250 symbol periods the instantaneous SINR at the output of the Dth stage of CGRRF is estimated according to (71) with $\mathbf{R}_{I+N}$ replaced by its estimate $\mathbf{R}_{I+N}(k) = \mathbf{R}(k) - \epsilon_1\mathbf{h}_1(k)\mathbf{h}_1^H(k)$.[5] The instantaneous SINR so obtained is time-averaged using the forgetting factor of 0.95. The value of rank (D) is then either increased or decreased by 1 in order to keep time-averaged SINR within the range 3 ± 1 dB.

In Figures 8 and 9, one realization of the time-averaged SINR and of the rank D versus time is shown. The propagation channel was fixed in this experiment in order to simplify the analysis. Initially, there are 16 users in the system with the desired user having the SNR of 8 dB and interferers at 14 dB. Starting rank value is 2. Over the first 1000 samples, the rank can be seen to converge and then stabilize at $D = 5$. At time $k = 4000$, six interfering users quit the system and D decreases to 3. Finally, at $k = 6000$, three of interferers reenter the communication and the filter rank again grows to 4.

Figure 8 shows that the reception quality is kept reasonably well within the required limits. Another advantage of this (or similar) adaptive rank selection techniques is the efficient utilization of the available processing power.

[5] The estimate of the desired user energy (ϵ_1) is supposed to be available at the receiver.

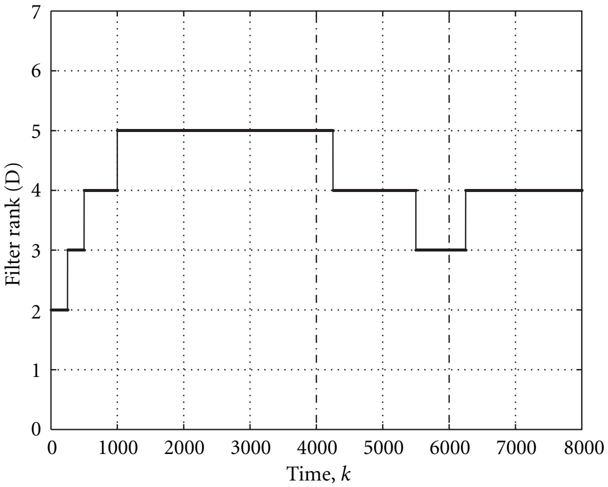

Figure 9: Rank D versus time for the CGRRF with adaptive rank selection.

6. CONCLUSIONS

A family of recently introduced least-square adaptive filtering techniques has been studied. All of these algorithms are shown to project and filter the received observation in the same low-dimensional subspace. This (Krylov) subspace is generated by taking the powers of the received covariance matrix on the cross-correlation (steering) vector. Consequently, considered techniques are related to Krylov subspace methods for linear systems and can be divided in four distinct groups: exact, auxiliary-vector, recursive, and approximate sample-by-sample methods. Numerical studies and complexity figures compare favorably the presented methods to conventional SMI and matched filtering techniques.

REFERENCES

[1] R. A. Monzingo and T. W. Miller, *Introduction to Adaptive Arrays*, Wiley, New York, NY, USA, 1980.

[2] S. Haykin, *Adaptive Filter Theory*, Prentice-Hall, Englewood-Cliffs, NJ, USA, 3rd edition, 1996.

[3] M. L. Honig and M. K. Tsatsanis, "Adaptive techniques for multiuser CDMA receivers," *IEEE Signal Processing Magazine*, vol. 17, no. 3, pp. 49–61, 2000.

[4] G.-O. Glentis, K. Berberidis, and S. Theodoridis, "Efficient least squares adaptive algorithms for FIR transversal filtering," *IEEE Signal Processing Magazine*, vol. 16, no. 4, pp. 13–41, 1999.

[5] M. L. Honig and W. Xiao, "Performance of reduced-rank linear interference suppression," *IEEE Transactions on Information Theory*, vol. 47, no. 5, pp. 1928–1946, 2001.

[6] 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, no. 7, pp. 2943–2959, 1998.

[7] S. Burykh and K. Abed-Meraim, "Multi-stage reduced-rank adaptive filter with flexible structure," in *European Signal Processing Conference*, Toulouse, France, September 2002.

[8] G. Dietl, M. D. Zoltowski, and M. Joham, "Reduced-rank equalization for EDGE via conjugate gradient implementation of multi-stage nested wiener filter," in *Proc. 54th IEEE Vehicular Technology Conference*, pp. 1912–1916, Atlantic City, NJ, USA, October 2001.

[9] M. L. Honig and W. Xiao, "Adaptive reduced-rank interference suppression with adaptive rank selection," in *Proc. Milcom 2000*, vol. 2, pp. 747–751, Los Angeles, Calif, USA, October 2000.

[10] D. A. Pados and S. N. Batalama, "Joint space-time auxiliary-vector filtering for DS/CDMA systems with antenna arrays," *IEEE Trans. Communications*, vol. 47, no. 9, pp. 1406–1415, 1999.

[11] D. A. Pados and G. N. Karystinos, "An iterative algorithm for the computation of the MVDR filter," *IEEE Trans. Signal Processing*, vol. 49, no. 2, pp. 290–300, 2001.

[12] S. Burykh and K. Abed-Meraim, "Reduced-rank adaptive filtering using Krylov subspace basis: a comparative study," in *Proc. 39th Allerton Conference on Communications, Control and Computing*, Monticello, Ill, USA, October 2001.

[13] W. Chen and U. Mitra, "Comparison of reduced rank detection schemes for CDMA systems," in *Proc. 2001 IEEE International Symposium on Information Theory*, p. 175, Washington, DC, USA, June 2001.

[14] S. Moshavi, E. G. Kanterakis, and D. L. Schilling, "Multistage linear receivers for DS-CDMA systems," *International Journal of Wireless Information Networks*, vol. 3, no. 1, pp. 1–17, 1996.

[15] P. A. Zulch, J. S. Goldstein, J. R. Guerci, and L. S. Reed, "Comparison of reduced-rank signal processing techniques," in *Proc. 32nd Asilomar Conference on Signals, Systems and Computers*, vol. 1, pp. 421–425, Pacific Grove, Calif, USA, November 1998.

[16] W. L. Myrick, M. D. Zoltowski, and J. S. Goldstein, "Low-sample performance of reduced-rank power minimization based jammer suppression for GPS," in *Proc. IEEE 6th International Symposium on Spread Spectrum Techniques and Applications*, pp. 93–97, Parsippany, NJ, USA, September 2000.

[17] Y. Saad, "An overview of Krylov subspace methods with applications to control problems," Tech. Rep. MTNS89, RIACS, NASA Ames Research Center, Moffet Field, Calif, USA, 1989, http://www.cs.umn.edu/~saad.

[18] G. H. Golub and C. F. Van Loan, *Matrix Computations*, The John Hopkins University Press, Baltimore, Md, USA, 1989.

[19] E. Polak, *Computational Methods in Optimization*, Academic Press, New York, NY, USA, 1971.

[20] W. Xiao and M. L. Honig, "Convergence and tracking of adaptive reduced-rank interference suppression algorithms," in *Proc. 34th Asilomar Conference on Signals, Systems and Computers*, pp. 1143–1147, Pacific Grove, Calif, USA, November 2000.

[21] L. S. Reed, J. D. Mallet, and L. E. Brennan, "Rapid convergence rate in adaptive arrays," *IEEE Trans. on Aerospace and Electronics Systems*, vol. 10, no. 6, pp. 853–863, 1974.

[22] R. Müller and S. Verdú, "Spectral efficiency of low-complexity multiuser detectors," in *Proc. IEEE International Symposium on Information Theory*, p. 439, Sorrento, Italy, June 2000.

[23] W. Xiao and M. L. Honig, "Convergence analysis of adaptive full rank and multi-stage reduced-rank interference suppression," in *34th Annual Conference on Information Sciences and Systems*, Princeton, NJ, USA, March 2000.

[24] P. S. Chang and A. N. Willson Jr., "Analysis of conjugate gradient algorithms for adaptive filtering," *IEEE Trans. Signal Processing*, vol. 48, no. 2, pp. 409–418, 2000.

[25] M. L. Honig and J. S. Goldstein, "Adaptive reduced-rank residual correlation algorithms for DS-CDMA interference suppression," in *Proc. 32nd IEEE Asilomar Conference on Signal, Systems and Computer*, vol. 2, pp. 1106–1110, Pacific Grove, Calif, USA, November 1998.

[26] I. N. Psaromiligkos and S. N. Batalama, "Finite data record maximum SINR adaptive space-time processing," in *Proc. 10th IEEE Workshop on Statistical Signal and Array Processing*, pp. 677–681, Pocono Manor, Pa, USA, August 2000.

[27] M. Haardt, A. Klein, R. Koehn, et al., "The TD-CDMA based UTRA TDD mode," *IEEE Journal on Selected Areas in Communications*, vol. 18, no. 8, pp. 1375–1385, 2000.

[28] Working Document towards ETR/SMG-50402, "Selection procedures for the choice of radio transmission technologies of the Universal Mobile Telecommunications System (UMTS)," Version 0.9.4, ETSI, May 1997.

[29] Radiocommunication study groups, "Guidelines for evaluation of radio transmission technologies for FPLMTS," Draft, ITU, April 1996.

[30] P. Dent, G. E. Bottomley, and T. Croft, "Jakes fading model revisited," *Electronics Letters*, vol. 29, no. 13, pp. 1162–1163, 1993.

Sergueï Burykh was born in Odessa, Ukraine. He received the B.S. degree from the Odessa State Polytechnic University, Odessa, in 1998 and the M.S. degree from the École Nationale Supérieure des Télécommunications (ENST) Paris, Paris, in 1999, all in electrical and computer engineering. Currently he is working towards the Ph.D. degree at the Signal and Image Processing Department of École Nationale Supérieure des Télécommunications Paris. His research interests are in statistical signal processing with applications to wireless communications and adaptive filtering.

Karim Abed-Meraim was born in 1967. He received the state engineering degree from École Polytechnique, Paris, France, in 1990, the state engineering degree from École Nationale Supérieure des Télécommunications (ENST), Paris, France, in 1992, the M.S. degree from Paris-Sud XI University, Orsay, France, in 1992, and the Ph.D. degree from the École Nationale Supérieure des Télécommunications (ENST), Paris, France, in 1995 (in the field of signal processing and communications). From 1995 to 1998, he has been one of the research staff at the Electrical Engineering Department of the University of Melbourne where he worked on several research project related to blind system identification for wireless communications, blind source separation, and array processing for communications, respectively. Currently he is an Associate Professor (since 1998) at the Signal and Image Processing Department of École Nationale Supérieure des Télécommunications (ENST). His research interests are in signal processing for communications and include system identification, multiuser detection, space-time coding, adaptive filtering and tracking, and array processing and performance analysis. Dr. Abed-Meraim is an IEEE member and an Associate Editor for the IEEE transactions on signal processing.