Fast Adaptive Super-Exponential Multistage Beamforming for Cellular Base-Station Transceivers with Antenna Arrays

Massimiliano (Max) Martone, *Member, IEEE*

***Abstract*— A new blind adaptive beamforming algorithm is introduced. We show how cumulants of the received signals can be used to obtain the weights of the beamformer that perform blind extraction. The method is based on a spatial interpretation of a deconvolution procedure known as the super-exponential algorithm. The basic block processing algorithm is attractive because it can be transformed in an efficient adaptive algorithm which exhibits good tracking capability. To prove the effectiveness of the idea, we show results for a typical mobile communications scenario where several cochannel inteferers corrupt the signals of interest.**

***Index Terms*—Array signal processing, higher order statistics, interference suppression, land mobile radio cellular systems.**

I. Introduction

THE USE OF antenna arrays in a communication system can theoretically improve performance in terms of capacity. Particularly, a multielement antenna receiver at the base station of a cellular communication system is able to compensate signal degradations in the mobile-to-base link caused by cochannel interference which is known to be the most important factor limiting the number of users that a system can handle. The traditional beamforming approach requires the knowledge of a "look" direction (the direction of arrival of the signal of interest) or the waveform of the signal of interest itself which is obviously not available in the cellular environment. Several alternative solutions have been proposed to solve the problem. The application of high-resolution array processing methods is not possible due to the extremely high number of wavefronts impinging over the array: the model is not identifiable. However, the application of subspace methods was proposed in [9], where the propagation model was considerably simplified assuming a local scattering mechanism. Blind adaptive beamforming methods appear to be more successful because no knowledge about array configuration, look direction, or desired signal is required. The most popular approach to blind beamforming is the constant modulus algorithm (CMA) array [12], [13], which represents the extension of the Godard blind equalization idea [16] to space filtering. The CMA method, which is basically very simple to implement, appeared to suffer from two main disadvantages: misconvergence and slow adaption rate. The application of a class of algorithms that exploit second-order cyclostationary properties of the received signals to separate the signal of interest from interferers was first presented in [17] motivated by the fact that many signals in communications are cyclostationary [4]. It is important also to mention the recent contribution of [18], where a method based on a gradient-based minimization of a new cost function was presented. The basic idea of that work is to exploit the property of cyclostationary signals to generate spectral lines when they pass through certain nonlinearities. Cumulant-based methods were presented in [6] to solve the blind beamforming problem, but no attempt was made to derive real-time adaptive algorithms. In [7], some cumulant-based methods were also introduced to show the advantages consequent to the use of higher order statistics. The method proposed in this work is based on the same idea introduced in [1] and [2], where the super-exponential approach [3] was generalized to the multivariate case. Here, we describe the application of the method to the space-only case and present a multistage implementation based on the architecture of [14]. The advantages of the proposed method are in the following facts.

- The approach is blind which allows the use of arbitrary array geometry and applicability in any propagation environment.
- It does not exhibit the typical problems of blind approaches to beamforming, in fact, it has the property of being globally convergent.
- The adaptive algorithm is sufficiently fast to track channel variations caused by moving transmitters, while at the same time being highly attractive from the computational point of view, proving that the use of higher order statistics does not necessarily imply slow convergence and, hence, extremely large sample size.

The paper is organized as follows. In Section II, we describe the discrete-time model for the communication system under analysis. In Section III, we describe the beamforming architecture, while the basic separating criterion to extract one of the signals is justified in Section IV. In Section V, a fully adaptive implementation is proposed, while in Section VI the results of some simulations for AMPS [21], [22], the current cellular system are shown.

Manuscript received September 23, 1996; revised September 25, 1998. This paper was presented in part at the International Conference on Wireless Communications'97, Calgary, Canada, July 1997.

The author is with the Telecommunications Group, Watkins-Johnson Company, Gaithersburg, MD 20878-1794 USA.

Publisher Item Identifier S 0018-9545(99)05734-5.

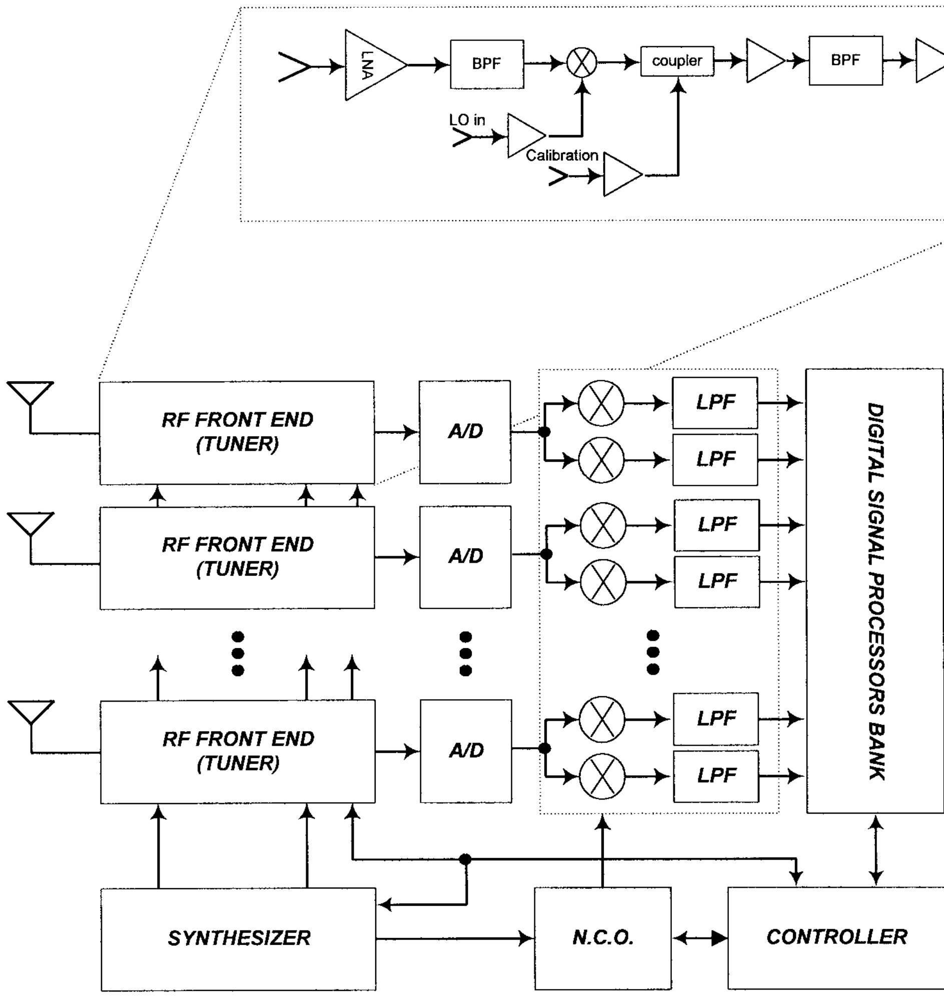

Fig. 1. Block diagram of the receiver.

II. System Model

We assume U mobile transmitters communicating with a base station with a K-element antenna, with $U \leq K$. A block diagram of the receiver is depicted in Fig. 1. The complex baseband-modulated signal transmitted by the lth transmitter is $x_l(t')$. We use the notation $s(t')$ to denote a continuous time waveform, while we will denote $s(t)$ as the discrete-time signal obtained by sampling at equispaced instants. Due to radio frequency (RF) multipath propagation (we will consider only the short-term fading which obeys a Rayleigh distribution), the signal received at the kth sensor of the array can be modeled as

$$r_k(t') = \sum_{l=1}^{U} \sum_{m=1}^{N_l} \rho_{l,m} e^{j\psi_{l,m}} a_k(\theta_{l,m}) x_l(t') + \eta_k(t') \qquad (1)$$

where N_l is the number of paths relative to the scattering model of the lth transmitter, $\rho_{l,m}$ is a Rayleigh-distributed random variable, $\psi_{l,m}$ is uniformly distributed over $[0, 2\pi]$, $a_k(\theta_{l,m})$ is the unknown gain and phase response of the kth sensor in the angle of arrival $\theta_{l,m}$, and $\eta_k(t')$ is the noise at the kth sensor. It is important to observe that the application of the algorithm described in the following is not limited to any array configuration, and it can be applied also when the narrow-band assumption does not hold. Sampling at $1/T_s$ rate, we can compact this expression as

$$r_k(tT_s) = y_k(t) = \sum_{l=1}^{U} h_{k,l} x_l(t) + \eta_k(t), \quad k = 1, 2, \cdots, K \qquad (2)$$

where $h_{k,l}$ is the T_s-sampled response of the array combined with the channel of the lth transmitted signal as seen at the kth sensor of the antenna array.[1]

III. Description of the Multistage Architecture

The concept of multistage separation is similar to the idea described in [14]. In this work, we use a new procedure to

[1] Observe that a beamforming method based on a high-resolution direction-finding approach requires $K \geq \Sigma_{l=1}^{U} N_l$ sensors according to the model (1). An accurate knowledge of the array geometry and the ability to resolve the wavefronts are also necessary. In [9], a reduced number of dominant paths was considered based on some geometrical assumptions. The blind approach overcomes the multipath modeling problem and requires only fewer sources (interferers) than sensors (motivated by a fundamental multivariable system theory limit). This assumption appears very reasonable since generally the frequency reuse planning of a cellular system is such that at a certain given time a small number of cochannel interferers has high enough power to degrade quality of service.

extract signals at every stage and show how, as a byproduct of the super-exponential method, we can eliminate the LMS search for the canceller weights of [14]. At each mth stage, we have to separate one of the U_m signals (the i_mth signal, where $i_m \in [1, U_m]$) using a K-element spatial filter (observe that $U_1 = U$ and $U_{m+1} = U_m - 1$). Generally, the permutation uncertainty inherent to the blind separation problem [8] causes the index i_m to be unknown *a priori*. We use a *no-noise* model for the derivation of the algorithm. Define $y_l^{(m)}(t), w_{i_m}^{(m)}, z_{i_m}^{(m)}(t)$ for $l = 1, 2, \cdots, K$ and $i_m = 1, 2 \cdots, U_m$ as inputs, weights, and outputs of the mth (beamformer) stage, respectively. Every stage has to solve a separation problem with a $K \times U_m$ mixing matrix, by extracting one of the sources. Then the contribution of the extracted signal to the array input is subtracted and a new separation problem with $U_m - 1$ sources is solved.

The output of the beamformer at the mth stage is $z_{i_m}^{(m)}(t) = \Sigma_{l=1}^{K} w_{i_m,l}^{(m)} y_l^{(m)}(t)$ for $i_m \in [1, U_m]$, where $w_{i_m,l}^{(m)}$ is the lth weight corresponding to the mth (stage) spatial filter designed to separate the i_m source. The overall input/output relation of the system including channel effects, array response, and space filter at the mth stage, but not including the additive noise is $z_{i_m}^{(m)}(t) = \Sigma_{l=1}^{U_m} s_{i_m,l}^{(m)} x_l(t)$, where $s_{i_m,l_1}^{(m)} = \Sigma_{l=1}^{K} w_{i_m,l}^{(m)} h_{l,l_1}^{(m)}$ for $i_m \in [1, U_m]$ (observe that $h_{i,l}^{(1)} = h_{i,l}$) or in a vector form[2]

$$\tilde{s}_{i_m}^{(m)} = \tilde{\boldsymbol{H}}_m \tilde{\boldsymbol{w}}_{i_m}^{(m)} \tag{3}$$

where $\tilde{\boldsymbol{H}}_m$, $\tilde{\boldsymbol{s}}_{i_m}^{(m)}$, and $\tilde{\boldsymbol{w}}_{i_m}^{(m)}$ are defined as

$$\begin{aligned}
[\tilde{\boldsymbol{H}}_m]_{l,k} &= h_{k,l}^{(m)} \\
\tilde{\boldsymbol{s}}_{i_m}^{(m)} &= [s_{i,1}^{(m)}, s_{i,2}^{(m)}, s_{i,3}^{(m)}, \cdots, s_{i,U_m}^{(m)}]^T \\
\tilde{\boldsymbol{w}}_{i_m}^{(m)} &= [w_{i_m,1}^{(m)}, w_{i_m,2}^{(m)}, \cdots, w_{i_m,K}^{(m)}]^T.
\end{aligned}$$

The *desired* response $\tilde{\boldsymbol{s}}_{i_m}^{(m)}$ (that is the response that separates the i_mth source) can be expressed as $\tilde{\boldsymbol{\delta}}_{i_m}^{(m)} = [\delta_{i_m,1}, \delta_{i_m,2}, \delta_{i_m,3}, \cdots, \delta_{i_m,U_m}]^T$, where

$$\delta_{i_m,k} = \begin{cases} 0, & i_m \neq k \\ 1, & i_m = k. \end{cases}$$

It is possible to solve the problem of finding the beamformer $\tilde{\boldsymbol{w}}_{i_m}^{(m)}$ that approximates the desired response solving the minimization problem

$$\min_{\tilde{\boldsymbol{w}}_{i_m}^{(m)}} ||\tilde{\boldsymbol{H}}_m \tilde{\boldsymbol{w}}_{i_m}^{(m)} - \tilde{\boldsymbol{\delta}}_{i_m}^{(m)}||^2. \tag{4}$$

The solution of (4) is

$$\tilde{\boldsymbol{w}}_{i_m}^{(m)} = (\tilde{\boldsymbol{H}}_m^H \tilde{\boldsymbol{H}}_m)^\dagger \tilde{\boldsymbol{H}}_m^H \tilde{\boldsymbol{\delta}}_{i_m}^{(m)}. \tag{5}$$

[2] $\boldsymbol{M}^H$ and $\boldsymbol{v}^H$, $\boldsymbol{M}^T$ and $\boldsymbol{v}^T$, and $\boldsymbol{M}^*$ and $\boldsymbol{v}^*$ denote complex conjugation transpose, transpose, and complex conjugation for the matrix $\boldsymbol{M}$ and vector $\boldsymbol{v}$, respectively. The k, l element of the matrix $\boldsymbol{M}$ and the mth element of the vector $\boldsymbol{v}$ are $[\boldsymbol{M}]_{k,l}$ and $[\boldsymbol{v}]_m$, respectively. Complex conjugation for a scalar a is identified by a^*. $||\boldsymbol{v}|| = \sqrt{\Sigma_{i=1}^{M} |[\boldsymbol{v}]_i|^2}$ is the 2-norm of the complex M vector.

and $\hat{\boldsymbol{s}}_{i_m}^{(m)} = \tilde{\boldsymbol{H}}_m (\tilde{\boldsymbol{H}}_m^H \tilde{\boldsymbol{H}}_m)^\dagger \tilde{\boldsymbol{H}}_m^H \tilde{\boldsymbol{\delta}}_{i_m}^{(m)}$, where we have used the symbol † to identify the inversion of a rank-deficient square matrix by, for example, singular value decomposition (SVD). Observe that this solution is exactly the optimum Wiener filter in presence of noise (up to a constant). In fact, the Wiener solution obtained minimizing the MSE is

$$\hat{\boldsymbol{w}}_{i_m} = E\{\tilde{\boldsymbol{y}}^{(m)}(k) \tilde{\boldsymbol{y}}^{(m)^H}(k)\}^\dagger E\{\tilde{\boldsymbol{y}}^{(m)}(k)^* x_{i_m}(k)\}. \tag{6}$$

Expression (6) is equivalent to (5) given that $E\{\tilde{\boldsymbol{y}}^{(m)}(k) \tilde{\boldsymbol{y}}^{(m)^H}(k)\} = \sigma_x^2 \tilde{\boldsymbol{H}}_m^H \tilde{\boldsymbol{H}}_m + \boldsymbol{\Gamma}_m$ with $[\tilde{\boldsymbol{\Gamma}}_m]_{l,n} = E\{\eta_n^{(m)}(k) \eta_l^{(m)^*}(k)\}$ and $E\{\tilde{\boldsymbol{y}}^{(m)}(k)^* x_{i_m}(k)\} = \sigma_x^2 \tilde{\boldsymbol{H}}_m^H \tilde{\boldsymbol{\delta}}_{i_m}^{(m)}$ [$\eta_n^{(m)}(k)$ is the additive noise at the mth stage, which was neglected in the derivation of (5) so that $\tilde{\boldsymbol{\Gamma}}_m = \boldsymbol{0}$].

A. Cumulants of Stationary Processes

If $\boldsymbol{x} = [x_1, x_2, \cdots, x_K]^T$ is a collection of random variables and $\boldsymbol{v} = [v_1, v_2, \cdots, v_K]^T$ is a collection of deterministic variables, then the Kth-order cumulant is defined as the Kth coefficient in the Mac–Laurin series expansion of the cumulant generating function

$$\mathcal{K}_x(\boldsymbol{v}) = ln E\{e^{j\boldsymbol{v}^T \boldsymbol{x}}\}. \tag{7}$$

Alternatively [11], one can define Kth-order cumulants as combinations of joint moments of orders up to K. Particularly for zero-mean random variables and $K = 2, 3, 4$

$$\begin{aligned}
\text{cum}\,[x_1^{(*)}, x_2^{(*)}] &= E\{x_1^{(*)} x_2^{(*)}\} \\
\text{cum}\,[x_1^{(*)}, x_2^{(*)}, x_3^{(*)}] &= E\{x_1^{(*)} x_2^{(*)} x_3^{(*)}\} \\
\text{cum}\,[x_1^{(*)}, x_2^{(*)}, x_3^{(*)}, x_4^{(*)}] &= E\{x_1^{(*)} x_2^{(*)} x_3^{(*)} x_4^{(*)}\} \\
&\quad - E\{x_1^{(*)} x_2^{(*)}\} E\{x_3^{(*)} x_4^{(*)}\} \\
&\quad - E\{x_1^{(*)} x_3^{(*)}\} E\{x_2^{(*)} x_4^{(*)}\} \\
&\quad - E\{x_1^{(*)} x_4^{(*)}\} E\{x_2^{(*)} x_3^{(*)}\}
\end{aligned}$$

where $x_l^{(*)}$ is an eventual complex conjugation for x_l. In this work the random variables are samples collected at time nT_s from the sensor outputs of an array. These collections of samples are modeled as stationary random processes. The fundamental properties of cumulants of random zero-mean stationary processes are as follows.

- *LIN:* $\text{cum}\,[\Sigma_i\, f_i x_i(n), \cdots] = \sigma_i f_i \text{cum}\,[\Sigma_i\, x_i(n), \cdots,]$.
- *STATIND: if the samples of a process can be divided into two (or more) statistically independent subsets, then their joint cumulants are zero.*
- *GAUSS: if the samples of a process are jointly Gaussian, then their joint K cumulant is zero for K greater than two.*

When the samples are well separated in time and if the cumulants are absolutely summable, then the theoretical cumulants are consistently estimated from a data record of N samples and ensemble averages can be approximated by empirical averages, simply exploiting the cumulant to moment equations.

The fundamental assumption necessary to develop the algorithm is as follows.

• **AS1:** The complex zero-mean discrete-time processes $\{x_n(t)\}$, for $n = 1, 2, \cdots, U$, are constituted by random variables identically non-Gaussian distributed. In addition, the cumulants of $\{x_n(t)\}$ satisfy:

1) $\text{cum}[x_k(t), x_l^*(t)] = \sigma_x^2 > 0$, only for $l = k$;
2) $\text{cum}[x_{n_1}(t), x_{n_2}^*(t), x_{n_3}(t), x_{n_4}^*(t)] = \gamma_{4x} \neq 0$, only for $n_1 = n_2 = n_3 = n_4$.

IV. Extraction of the i_m Signal (mth Stage)

In this section, we will show how to obtain an estimate of the optimal solution for the weights of the beamformer and at the same time the propagation vector for the i_mth signal. The following two-step iterative procedure defines a class of algorithms for different values of $\tilde{p}$ and $\tilde{q}$ ($\tilde{p} + \tilde{q} \geq 2$) [1], [3]:

$$\texttt{Step 1} \Rightarrow s_{i_m,l}^{(m)} = (s_{i_m,l}^{(m)})^{\tilde{p}} (s_{i_m,l}^{(m)^*})^{\tilde{q}} \tag{8}$$

$$\texttt{Step 2} \Rightarrow s_{i_m,l}^{(m)} = s_{i_m,l}^{(m)} \frac{1}{\sqrt{\sum_{l=1}^{U_m} |s_{i_m,l}^{(m)}|^2}}. \tag{9}$$

Choosing $\tilde{p} = 2$, $\tilde{q} = 1$ gives a solution in terms of fourth- and second-order cumulants. As observed in [3], (8) and (9) operated on $\tilde{\boldsymbol{s}}_{i_m}^{(m)}$ converges at a *super-exponential rate* to the desired solution $\tilde{\boldsymbol{\delta}}_{i_m}$. Since obviously $\tilde{\boldsymbol{s}}_{i_m}^{(m)}$ is not available (because $\tilde{\boldsymbol{H}}_m$ is not known), we derive a procedure in terms of $\tilde{\boldsymbol{w}}_{i_m}^{(m)}$. If we define $\tilde{\boldsymbol{g}}_{i_m}^{(m)} = [g_{i_m,1}^{(m)}, g_{i_m,2}^{(m)}, g_{i_m,3}^{(m)}, \cdots, g_{i_m,U_m}^{(m)}]^T$ as the vector obtained by $g_{i_m,l}^{(m)} = (s_{i_m,l}^{(m)})^2 s_{i_m,l}^{(m)^*}$, we can state the least squares minimization problem

$$\min_{\tilde{\boldsymbol{w}}_{i_m}^{(m)}} \|\tilde{\boldsymbol{H}}_m \tilde{\boldsymbol{w}}_{i_m}^{(m)} - \tilde{\boldsymbol{g}}_{i_m}^{(m)}\|^2 \tag{10}$$

with the solution

$$\tilde{\boldsymbol{w}}_{i_m}^{(m)} = (\tilde{\boldsymbol{H}}_m^H \tilde{\boldsymbol{H}}_m)^\dagger \tilde{\boldsymbol{H}}_m^H \tilde{\boldsymbol{g}}_{i_m}^{(m)}. \tag{11}$$

To obtain normalization (9), the second step is

$$\tilde{\boldsymbol{w}}_{i_m}^{(m)} = \frac{\tilde{\boldsymbol{w}}_{i_m}^{(m)}}{\sqrt{\tilde{\boldsymbol{w}}_{i_m}^{(m)^H} (\tilde{\boldsymbol{H}}_m^H \tilde{\boldsymbol{H}}_m) \tilde{\boldsymbol{w}}_{i_m}^{(m)}}}. \tag{12}$$

The algorithm in the $\tilde{\boldsymbol{w}}_{i_m}$ domain [see (11) and (12)] *projected* back in the $\tilde{\boldsymbol{s}}_{i_m}$ domain becomes

$$\tilde{\boldsymbol{s}}_{i_m}^{(m)} = \tilde{\boldsymbol{H}}_m (\tilde{\boldsymbol{H}}_m^H \tilde{\boldsymbol{H}}_m)^\dagger \tilde{\boldsymbol{H}}_m^H \tilde{\boldsymbol{g}}_{i_m}^{(m)} \quad \tilde{\boldsymbol{s}}_{i_m}^{(m)} = \frac{\tilde{\boldsymbol{s}}_{i_m}^{(m)}}{\|\tilde{\boldsymbol{s}}_{i_m}^{(m)}\|} \tag{13}$$

whose point of convergence easily obtained (see [3]) is

$$\tilde{\boldsymbol{s}}_{i_m}^{(m)} \simeq \frac{\tilde{\boldsymbol{H}}_m (\tilde{\boldsymbol{H}}_m^H \tilde{\boldsymbol{H}}_m)^\dagger \tilde{\boldsymbol{H}}_m^H \tilde{\boldsymbol{\delta}}_{i_m}^{(m)}.}{\sqrt{\tilde{\boldsymbol{\delta}}_{i_m}^{(m)^T} \tilde{\boldsymbol{H}}_m (\tilde{\boldsymbol{H}}_m^H \tilde{\boldsymbol{H}}_m)^\dagger \tilde{\boldsymbol{H}}_m^H \tilde{\boldsymbol{\delta}}_{i_m}^{(m)}}}.$$

This expression is coincident with the solution (5) up to a gain factor. In Appendix A, we further explain the super-exponential algorithm and the related convergence issues.

The procedure [see (11) and (12)] can be expressed in terms of the cumulants of the outputs of the sensors. We exploit the above assumptions and the properties of the cumulants of linear stationary processes (see [11]) so that we can write

$$\begin{aligned} &\text{cum}\,[y_{i_1}^{(m)}(n), y_{i_2}^{(m)^*}(n)] \\ &\quad = \text{cum}\left[\sum_{l_1=1}^{U_m} h_{i_1,l_1}^{(m)} x_{l_1}(n), \sum_{l_2=1}^{U_m} h_{i_2,l_2}^* x_{l_2}^*(n)\right] \\ &\quad = \sum_{l=1}^{U_m} h_{i_1,l}^{(m)} h_{i_2,l}^* \sigma_x^2 \end{aligned} \tag{14}$$

due to

$$\text{cum}\,[x_{l_1}(k), x_{l_2}^*(k)] = \begin{cases} \text{cum}\,[x_l(n), x_l^*(n)] = \sigma_x^2, & l_1 = l_2 \\ 0, & \text{otherwise.} \end{cases}$$

To derive the second key expression related to (11), let us consider

$$\begin{aligned} &\text{cum}\,[z_{i_m}^{(m)}(k), z_{i_m}^{(m)}(k), z_{i_m}^{(m)^*}(k), y_{i_3}^{(m)^*}(k)] \\ &\quad = \sum_{l=1}^{U_m} h_{i_3,l}^* \,\text{cum}[z_{i_m}^{(m)}(k), z_{i_m}^{(m)}(k), z_{i_m}^{(m)^*}(k), x_l^*(k)] \end{aligned} \tag{15}$$

and

$$\begin{aligned} &\text{cum}\,[z_{i_m}^{(m)}(k), z_{i_m}^{(m)}(k), z_{i_m}^{(m)^*}(k), x_l^*(k)] \\ &\quad = \sum_{l_1=1}^{U_m} \sum_{l_2=1}^{U_m} \sum_{l_3=1}^{U_m} s_{i_m,l_1}^{(m)} s_{i_m,l_2}^{(m)} s_{i_m,l_3}^{(m)^*} \\ &\qquad \times \text{cum}\,[x_{l_1}(k), x_{l_2}(k), x_{l_3}^*(k), x_l^*(k)] \\ &\quad = \gamma_{4x} s_{i_m,l}^{(m)^2} s_{i_m,l}^{(m)^*} = \gamma_{4x} g_{i_m,l}^{(m)} \end{aligned} \tag{16}$$

where the last equality follows from

$$\text{cum}\,[x_{l_1}(n), x_{l_2}(n), x_{l_3}^*(n), x_l^*(n)] = \begin{cases} \gamma_{4x}, & l_1 = l_2 = l_3 = l \\ 0, & \text{otherwise.} \end{cases}$$

So we can write

$$\begin{aligned} &\text{cum}\,[z_{i_m}^{(m)}(k), z_{i_m}^{(m)}(k), z_{i_m}^{(m)^*}(k), y_{i_3}^{(m)^*}(k)] \\ &\quad = \sum_{l=1}^{U_m} h_{i_3,l}^{(m)^*} g_{i_m,l}^{(m)} \gamma_{4x}. \end{aligned} \tag{17}$$

Expressions (14) and (17) can be substituted in (11), and the following iterative algorithm is obtained:

$$\texttt{Step 1} \Rightarrow \tilde{\boldsymbol{w}}_{i_m}^{(m)} = \tilde{\boldsymbol{R}}^{(m)^\dagger} \boldsymbol{D}_{i_m}^{(m)} \tag{18}$$

$$\texttt{Step 2} \Rightarrow \tilde{\boldsymbol{w}}_{i_m}^{(m)} = \frac{\tilde{\boldsymbol{w}}_{i_m}^{(m)}}{\sqrt{\tilde{\boldsymbol{w}}_{i_m}^{(m)^H} \tilde{\boldsymbol{R}}^{(m)} \tilde{\boldsymbol{w}}_{i_m}^{(m)}}} \tag{19}$$

where the generic i, l element of the $K \times K$ matrix $\tilde{\boldsymbol{R}}^{(m)}$ is given by

$$[\tilde{\boldsymbol{R}}^{(m)}]_{i,l} = \frac{\text{cum}[y_l^{(m)}(k), y_i^{(m)^*}(k)]}{\sigma_x^2} \tag{20}$$

and the lth element of the $K \times 1$ vector $\boldsymbol{D}_{i_m}^{(m)}$ of fourth-order cumulants is given by

$$[\boldsymbol{D}_{i_m}^{(m)}]_l = \frac{\text{cum}[z_{i_m}^{(m)}(k), z_{i_m}^{(m)}(k), z_{i_m}^{(m)^*}(k), y_l^{(m)^*}(k)]}{\gamma_{4x}}. \quad (21)$$

Now if we take into account the additive noise $\eta_l^{(m)}(k)$, we have

$$\tilde{\boldsymbol{R}}^{(m)} = \frac{1}{\sigma_x^2}[\sigma_x^2 \tilde{\boldsymbol{H}}_m^H \tilde{\boldsymbol{H}}_m + \tilde{\boldsymbol{\Gamma}}_m] \quad (22)$$

and $\boldsymbol{D}_{i_m}^{(m)} \simeq \tilde{\boldsymbol{H}}_m^H \tilde{\boldsymbol{g}}_{i_m}^{(m)}$ because

$$\begin{aligned} &\text{cum}[z_{i_m}^{(m)}(k), z_{i_m}^{(m)}(k), z_{i_m}^{(m)^*}(k), y_{i_3}^{(m)^*}(k)] \\ &\quad = \sum_{l=1}^{U_m} h_{i_3,l}^{(m)^*} g_{i_m,l}^{(m)} \gamma_{4x} \\ &\quad\quad + \sum_{l_1} \sum_{l_2} \sum_{l_3} w_{i_m,l_1}^{(m)} w_{i_m,l_2}^{(m)} w_{i_m,l_3}^{(m)^*} \\ &\quad\quad \times \text{cum}[\eta_{l_1}^{(m)}(k), \eta_{l_2}^{(m)}(k), \eta_{l_3}^{(m)^*}(k), \eta_{i_3}^{(m)^*}(k)] \\ &\quad = \sum_{l=1}^{U_m} h_{i_3,l}^{(m)^*} g_{i_m,l}^{(m)} \gamma_{4x}. \end{aligned} \quad (23)$$

In fact, $\eta_l^{(m)}(k)$ is a Gaussian process and its cumulants of an order greater than two vanish. So if the iterative algorithm converges close to the desired response so that $\tilde{\boldsymbol{g}}_{i_m} \simeq \text{const}\ \tilde{\boldsymbol{\delta}}_{i_m}$, then we have

$$\tilde{\boldsymbol{w}}_{i_m}^{(m)} \simeq \sigma_x^2 [\sigma_x^2 \tilde{\boldsymbol{H}}_m^H \tilde{\boldsymbol{H}}_m + \tilde{\boldsymbol{\Gamma}}_m]^\dagger \tilde{\boldsymbol{H}}_m^H \text{const}\ \tilde{\boldsymbol{\delta}}_{i_m}$$

which is exactly the optimal Wiener solution given in (6).

V. Adaptive Implementation

In this section, we derive an adaptive algorithm for online computation of the spatial filter weights. The derivation is based on the theory of recursive least squares (RLS). Sample statistics-based estimation of the cumulants of interest are

$$\overline{\text{cum}}[y_i(n); y_j^*(n)] = \frac{1}{N} \sum_{n=1}^{N} y_i(n) y_j^*(n) \quad (24)$$

$$\begin{aligned} &\overline{\text{cum}}[z_i(n); z_i(n); z_i^*(n); y_j^*(n)] \\ &\quad = \frac{1}{N} \sum_{n=1}^{N} |z_i(n)|^2 y_j^*(n) z(t) \\ &\quad - 2\frac{1}{N} \sum_{n=1}^{N} |z_i(n)|^2 \frac{1}{N} \sum_{n=1}^{N} z_i(n) y_j^*(n) \\ &\quad - \frac{1}{N} \sum_{n=1}^{N} z_i(n)^2 \frac{1}{N} \sum_{n=1}^{N} z_i^*(n) y_j^*(n). \end{aligned} \quad (25)$$

We have neglected stage indexes for $y_j(t), z_i(t)$ for simplicity and have indicated the estimated cumulant as $\overline{\text{cum}}[\cdot\cdot]$. At the end of the convergence process, the following equation must be satisfied:

$$\boldsymbol{D}_{i_m}^{(m)} - \tilde{\boldsymbol{R}}^{(m)} \tilde{\boldsymbol{w}}_{i_m}^{(m)} = \mathbf{0}. \quad (26)$$

TABLE I
THE PROPOSED ALGORITHM

- **Time update recursions:**
  1. $z_{i_m}^{(m)}(n) = \sum_{i=1}^{K} w_{i_m,i}^{(m)}(n-1) y_i^{(m)}(n)$
  2. $\beta_{i_m}^{(m)}(n) = \lambda \beta_{i_m}^{(m)}(n-1) + z_{i_m}^{(m)^2}(n)$
  3. $\tilde{z}_{i_m}^{(m)}(n) = \left[\left|z_{i_m}^{(m)}(n)\right|^2 - 2\,\sigma_x^2\right] z_{i_m}^{(m)}(n) - \beta_{i_m}(n) z_{i_m}^*(n)$
  4. $\mathbf{D}_{i_m}^{(m)}(n) = \lambda \mathbf{D}_{i_m}^{(m)}(n-1) + \tilde{\mathbf{y}}^{(m)^*}(n) \tilde{z}_{i_m}^{(m)}(n)/\gamma_{4x}$
  5. $\mathbf{P}^{(m)}(k) = \lambda^{-1}\left[\mathbf{P}^{(m)}(k-1) + \frac{\mathbf{P}^{(m)}(k-1)\tilde{\mathbf{y}}^{(m)^*}(k)\tilde{\mathbf{y}}^{(m)^T}(k)\mathbf{P}^{(m)}(k-1)}{\lambda + \tilde{\mathbf{y}}^{(m)^T}(k)\,\mathbf{P}^{(m)}(k-1)\tilde{\mathbf{y}}^{(m)^*}(k)}\right]$
  6. $\mathbf{K}^{(m)}(k) = \mathbf{P}^{(m)}(k) \tilde{\mathbf{y}}^{(m)^*}(k)$
  7. $\tilde{\mathbf{w}}_{i_m}^{(m)}(k) = \tilde{\mathbf{w}}_{i_m}^{(m)}(k-1) + \mathbf{K}^{(m)}(k)\left[\tilde{z}_{i_m}^{(m)}(k) - \tilde{\mathbf{y}}^{(m)^T}(k)\, \tilde{\mathbf{w}}_{i_m}^{(m)}(k-1)\right]$
- **Stage update recursions:**
  1. $\tilde{\mathbf{y}}^{(m+1)}(n) = \tilde{\mathbf{y}}^{(m)}(n) - \mathbf{D}_{i_m}^{(m)^*}(n) \tilde{z}_{i_m}^{(m)}(n)$
- **Initialization:**
  1. for $i = 1, 2, ..., K$: for $m = 1, 2, ..., U$ initialize $w_{i_m,i}^{(m)}(k_0)$
  2. for $m = 1, 2, ..., U$ initialize $\mathbf{D}_{i_m}^{(m)}(k_0)$
  3. for for $m = 1, 2, ..., U$ initialize $\mathbf{P}^{(m)}(k_0)$
  4. for $i = 1, 2..., K$: $y_i^{(1)}(k_0) = y_i(k_0)$,
  5. for $m = 1, 2, ...U$:
     - $z_{i_m}^{(m)}(k_0) = \sum_{i=1}^{K} w_{i_m,i}^{(m)}(k_0) y_i^{(m)}(k_0)$,
     - $y_i^{(m+1)}(k_0) = y_i^{(m)}(k_0) - \mathbf{D}_{i_m}^{(m)^*}(k_0)\ z_{i_m}^{(m)}(k_0)$,

The estimate of $\tilde{\boldsymbol{R}}^{(m)}$, $\boldsymbol{D}_{i_m}^{(m)}, \tilde{\boldsymbol{w}}_{i_m}^{(m)}$ based on k samples is $\tilde{\boldsymbol{R}}^{(m)}(k)$, $\boldsymbol{D}_{i_m}^{(m)}(k)$, $\tilde{\boldsymbol{w}}_{i_m}^{(m)}(k)$, and their recursive estimation can be obtained as

$$\tilde{\boldsymbol{R}}^{(m)}(k) = \lambda \tilde{\boldsymbol{R}}^{(m)}(k-1) + \frac{\tilde{\boldsymbol{y}}^{(m)^*}(k) \tilde{\boldsymbol{y}}^{(m)^T}(k)}{\sigma_x^2} \quad (27)$$

$$\boldsymbol{D}_{i_m}^{(m)}(k) = \lambda \boldsymbol{D}_{i_m}^{(m)}(k-1) + \frac{\tilde{\boldsymbol{y}}^{(m)^*}(k) \tilde{z}_{i_m}^{(m)}(k)}{\gamma_{4x}} \quad (28)$$

where

$$\begin{aligned} \tilde{\boldsymbol{y}}^{(m)}(k) &= [y_1^{(m)}(k), y_2^{(m)}(k), \cdots, y_K^{(m)}(k)]^T \\ \tilde{z}_{i_m}^{(m)}(k) &= (|z_{i_m}^{(m)}(n)|^2 - 2\sigma_x^2) z_{i_m}^{(m)}(n) - \beta_{i_m}(n) z_{i_m}^*(n) \\ \beta_{i_m}^{(m)}(k) &= \lambda \beta_{i_m}^{(m)}(k-1) + z_{i_m}^{(m)^2}(k). \end{aligned}$$

λ is the *forgetting factor*. The process $\beta_{i_m}^{(m)}(t)$ is the recursive estimation of $E\{z_{i_m}^{(m)^2}(k)\}$. Expression (28) can be justified by considering the estimation of fourth-order cumulants, based on sample averages given by (25) and the statistical assumptions on the process $x_l(n)$. Due to the power normalization $E\{|z_{i_m}^{(m)}(k)|^2\} = E\{z_{i_m}^{(m)}(k) z_{i_m}^{(m)^*}(k)\} = \sum_j |s_{i_m,j}^{(m)}|^2 E\{x_{i_m}(k) x_{i_m}^*(k)\} = \sigma_x^2$. Since we need the inverse of the correlation matrix at every step, we can use the matrix inverse identify and write the equation given at the bottom of the next page, with $\boldsymbol{P}^{(m)}(k) = (\tilde{\boldsymbol{R}}^{(m)}(k))^{-1}$. The Kalman gain is given by $\boldsymbol{K}^{(m)}(k) = \boldsymbol{P}^{(m)}(k) \tilde{\boldsymbol{y}}^{(m)^*}(k)$, and the recursive updating of the deconvolution filter is calculated as

$$\begin{aligned} \tilde{\boldsymbol{w}}_{i_m}^{(m)}(k+1) &= \tilde{\boldsymbol{w}}_{i_m}^{(m)}(k) + \boldsymbol{K}^{(m)}(k) \\ &\quad \cdot [\tilde{z}_{i_m}^{(m)}(k) - \tilde{\boldsymbol{y}}^{(m)^T}(k+1) \tilde{\boldsymbol{w}}_{i_m}^{(m)}(k)]. \end{aligned} \quad (29)$$

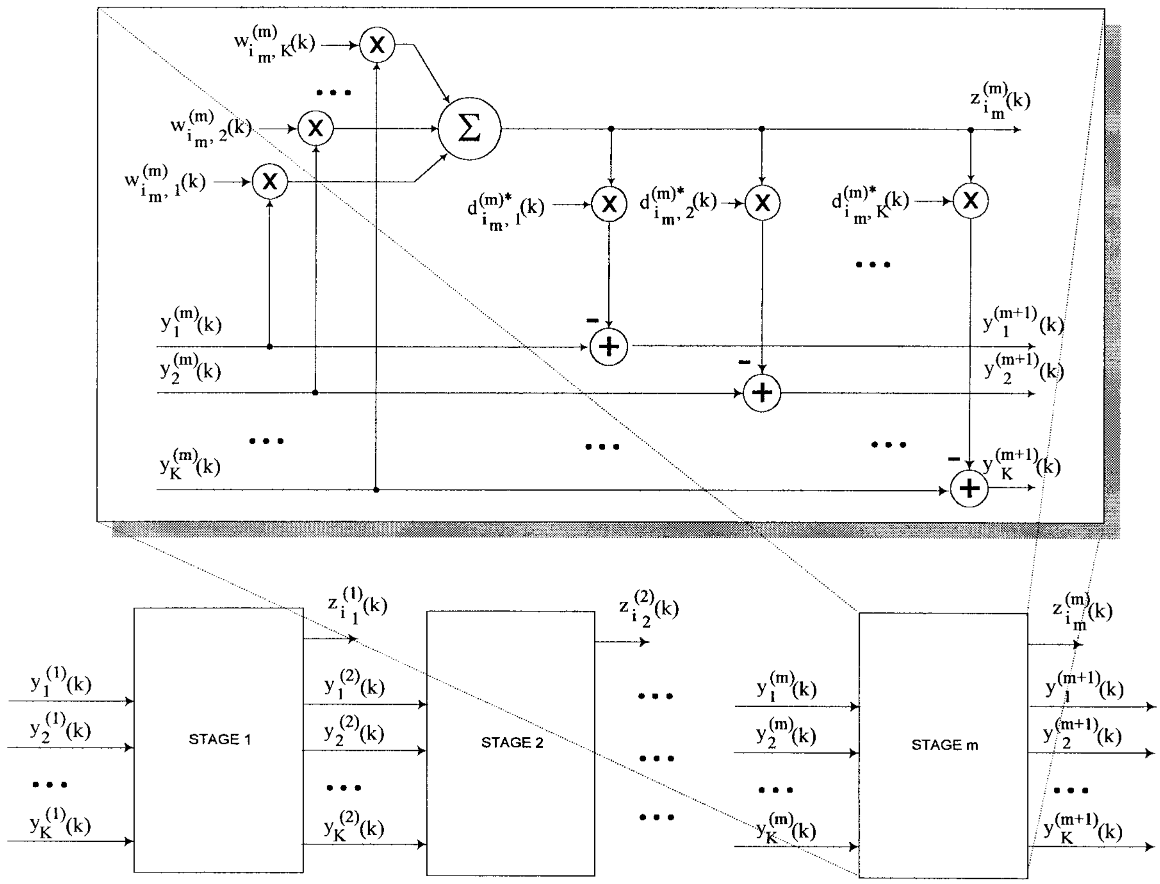

Fig. 2. Discrete-time model of the filtering section (K sensors up to the mth stage).

The initialization can be made by estimating cumulants using sample averages on a small number of data samples; the normalization required by the general method can be implemented by scaling the value of the weights obtained at each iteration.

Since $z_{i_m}^{(m)}(k) \overset{m.s.s.}{\rightarrow} x_{i_m}(k)$ (where $\overset{m.s.s.}{\rightarrow}$ stands for "converges in the mean-square sense") and as $\lim_{k\rightarrow\infty} \tilde{\boldsymbol{w}}_{i_m}^{(m)}(k) = C\hat{\boldsymbol{w}}_{i_m}^{(m)}$ (for some scalar C), it follows as a byproduct of the iterative-recursive procedure that

$$\begin{aligned}\lim_{k\rightarrow\infty} \boldsymbol{D}_{i_m}^{(m)}(k) &= C'\tilde{\boldsymbol{H}}_m^H\tilde{\boldsymbol{\delta}}_{i_m}^{(m)} \\ &= C'[\hat{h}_{1,i_m}^{(m)^*}, \hat{h}_{2,i_m}^{(m)^*}, \cdots, \hat{h}_{K,i_m}^{(m)^*}]^T.\end{aligned}$$

That is, the vector $\boldsymbol{D}_{i_m}^{(m)}(k)$ contains the information relative to the i_mth-directional vector up to a scalar C'. As a consequence, the stage update recursion is given by

$$\tilde{\boldsymbol{y}}^{(m+1)}(k) = \tilde{\boldsymbol{y}}^{(m)}(k) - \boldsymbol{D}_{i_m}^{(m)^*}(k)\tilde{z}_{i_m}^{(m)}(k).$$

The algorithm can be summarized as in Table I (see also Fig. 2). The lag k_0 can be selected arbitrarily, depending on the initialization strategy. It is well known that the updating of the matrix $\boldsymbol{P}^{(m)}(k)$ can become numerically unstable. A number of solutions can be employed to avoid this problem. We have studied and simulated a square root (QR decomposition) approach similar to the method proposed in [10], [19], and [20]. The algorithm is described in Appendix B.

VI. Simulations

As an example of application of the algorithm, we simulated the AMPS cellular system environment [21], [22]. A base station equipped with a $K = 10$ element uniform linear array with half-wavelength element spacing is considered. There are $U = 4$ signals impinging over the array with equal power to represent a rather pessimistic interference scenario. The number of rays in (1) to generate the fading channels is $N_1 = N_2 = N_3 = N_4 = 25$, while the transmitters' angles of arrival in the case of static channel are clustered for each path around 60°, 10°, −25°, and 30°, respectively, as described in [9] with $\theta_{BW} = 3°$. The signals are assumed to be received with equal strength. Carrier separation is 0 Hz for the RF-modulated signals. The sampling frequency is 80 KHz. White Gaussian noise afflicts all the signals impinging over the array with equal power.

$$\boldsymbol{P}^{(m)}(k+1) = \lambda^{-1}\left[\boldsymbol{P}^{(m)}(k) + \frac{\boldsymbol{P}^{(m)}(k)\tilde{\boldsymbol{y}}^{(m)^*}(k+1)\tilde{\boldsymbol{y}}^{(m)^T}(k+1)\boldsymbol{P}^{(m)}(k)}{\lambda + \tilde{\boldsymbol{y}}^{(m)^T}(k+1)\boldsymbol{P}^{(m)}(k)\tilde{\boldsymbol{y}}^{(m)^*}(k+1)}\right]$$

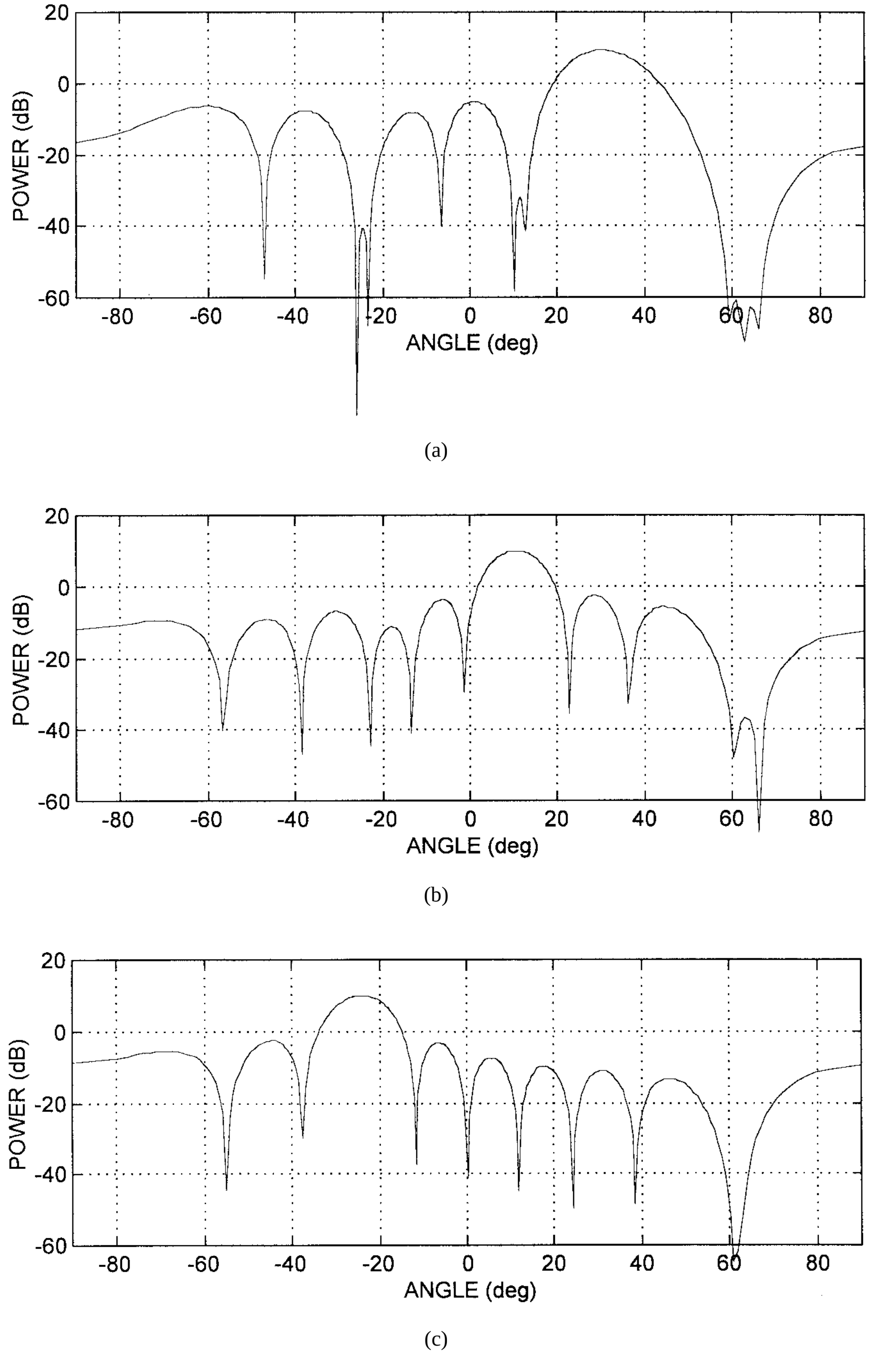

Fig. 3. Beam pattern of the first three stages: DOA's are $\theta_1 = 60°$, $\theta_2 = 10°$, $\theta_3 = -25°$, and $\theta_4 = 30°$. (a) The first stage captures θ_4 and forms nulls in the directions θ_2, θ_1, and θ_3. (b) The second stage captures θ_2 and forms nulls in the directions θ_3 and θ_1. (c) The third stage captures θ_3 and forms a null in the direction θ_1.

The fading channel is static in a first set of simulation experiments and it is generated randomly at every run according to the Rayleigh distribution. In Fig. 3, the beam pattern is shown for the four stages after 1000 samples, with $\lambda = 0.99$, with a signal-to-noise ratio (SNR) $= 30$ dB. In Fig. 4, the mean-squared error (MSE) of the four sources $\text{MSE}_1, \text{MSE}_2, \text{MSE}_3$, and MSE_4 is reported versus the number of samples processed where

$$\text{MSE}_{i_m} = E\{|x_{i_m}(k) - z_{i_m}(k)|^2\} \quad (30)$$

and the expectation is calculated by averaging over 100 independent runs. The weight updating algorithm is the QR recursive algorithm described in the Appendix with $\lambda = 0.855$.

Fig. 5 shows a comparison in terms of MSE of the first three captured sources (among the four of interest) with the multistage CMA array of [14]. The value of λ is 0.97.

The output-signal-to-interference-plus-noise ratio is defined as

$$\text{OSNIR}_{i_m} = E\left\{\left|\sum_{n=1}^{10} h_{n,i_m}^{(m)} w_{n,i_m}^{(m)} x_{i_m}(k)\right|^2\right\}(\tilde{N} + \tilde{I})^{-1}$$

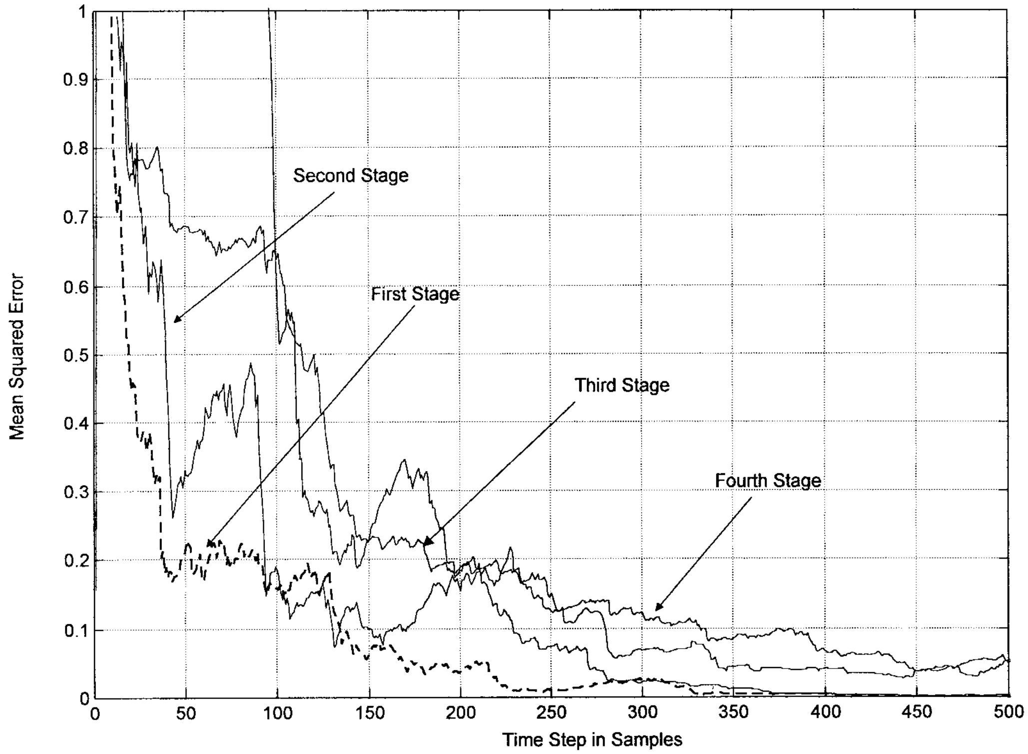

Fig. 4. Convergence process (same conditions as Fig. 3) for the four stages in terms of MSE.

where

$$\tilde{N} = \sum_{n=1}^{10} |w_{n,i_m}^{(m)}|^2 E\{\eta_l(k)\eta_l^*(k)\} \quad \text{and}$$

$$\tilde{I} = \sum_{\substack{i=1 \\ i \neq i_m}}^{U_m} E\left\{\left|\sum_{n=1}^{10} h_{n,i}^{(m)} w_{n,i_m}^{(m)} x_i(k)\right|^2\right\}$$

and the expectations are computed averaging over independent computer runs.

The tracking performance of the algorithm was tested in a simulation experiment and using a time-varying multipath channel. The Doppler frequency usually describes the second-order statistics of channel variations. Doppler frequency is related through wavelength $\tilde{\lambda}$ to vehicle motion. The model used in this case is based on the wide-sense stationary uncorrelated scattering (WSSUS) assumption [23], [24]. The complex baseband channel variations are generated as filtered Gaussian processes fully specified by the scattering function. In particular, each process has a frequency response equal to the square root of the Doppler power density spectrum. We approximated the Doppler spectrum by rational filtered processes. The filters are described by their 3-dB bandwidth which is called the normalized Doppler frequency. Figs. 6 and 7 show tracking performance of the algorithm for mobiles transmitting so that the maximum Doppler frequency (defined as $f_D = V/\tilde{\lambda}$, where V is vehicle speed and $\tilde{\lambda}$ is carrier wavelength) multiplied with the sampling period T_s is $f_D T_s = 0.0001$ (Fig. 6) and $f_D T_s = 0.0006$ (Fig. 7). The evolution of the OSNIR for the first three stages is shown in Fig. 6 for a $\lambda = 0.855$. In Fig. 7, performance on a typical fade event for $p_1(k)$ (power of the output of the first stage)[3] and SNR $= 30$ dB is shown. The forgetting factor is $\lambda = 0.97$. The real-time power at the output of the array is shown with respect to the optimum solution computed assuming perfect knowledge of the propagation environment.

VII. Conclusions

We have studied a new solution to blind beamforming for fourth-order white stationary sources. The algorithm is based on a generalization [1], [2] to space filtering of the idea presented in [3]. The method appears to converge rapidly to the optimum array response using an adaptive QR-based RLS approach. There is an increase in complexity with respect to the extreme simplicity of a traditional gradient-based search like the CMA array, but certainly significant computational savings with respect to high-resolution direction-finding algorithms. Moreover the algorithm does not exhibit the typical problems of blind approaches to beamforming while maintaining all the known benefits (for example, unknown array geometry and unsupervised operation). In fact, the adaptive algorithm has the property of being globally convergent and sufficiently fast to track channel variations caused by moving transmitters, proving that the use of higher order statistics does

[3] The power for the i_mth transmitter at the mth stage can be estimated using [14] $p_{i_m}(k) = |\tilde{\boldsymbol{w}}_{i_m}^{(m)^H}(k)[\Pi_{l=1}^{m-1}(\boldsymbol{I}_K - \boldsymbol{D}_{i_l}^{(l)^*}(k)\tilde{\boldsymbol{w}}_{i_l}^{(l)^H}(k))]\boldsymbol{h}_{i_m}(k)x_{i_m}(k)|^2$, where $\boldsymbol{h}_{i_m}(k) = [h_{1,i_m}^{(1)}(k), h_{2,i_m}^{(1)}(k), \cdots, h_{K,i_m}^{(1)}(k)]^T$ and $h_{i,l}^{(m)}(k)$ is the i,l element of the time-varying propagation matrix $\boldsymbol{H}_1(k)$ at time step k defined exactly as $\boldsymbol{H}_m$ for $m = 1$ in the time-invariant case.

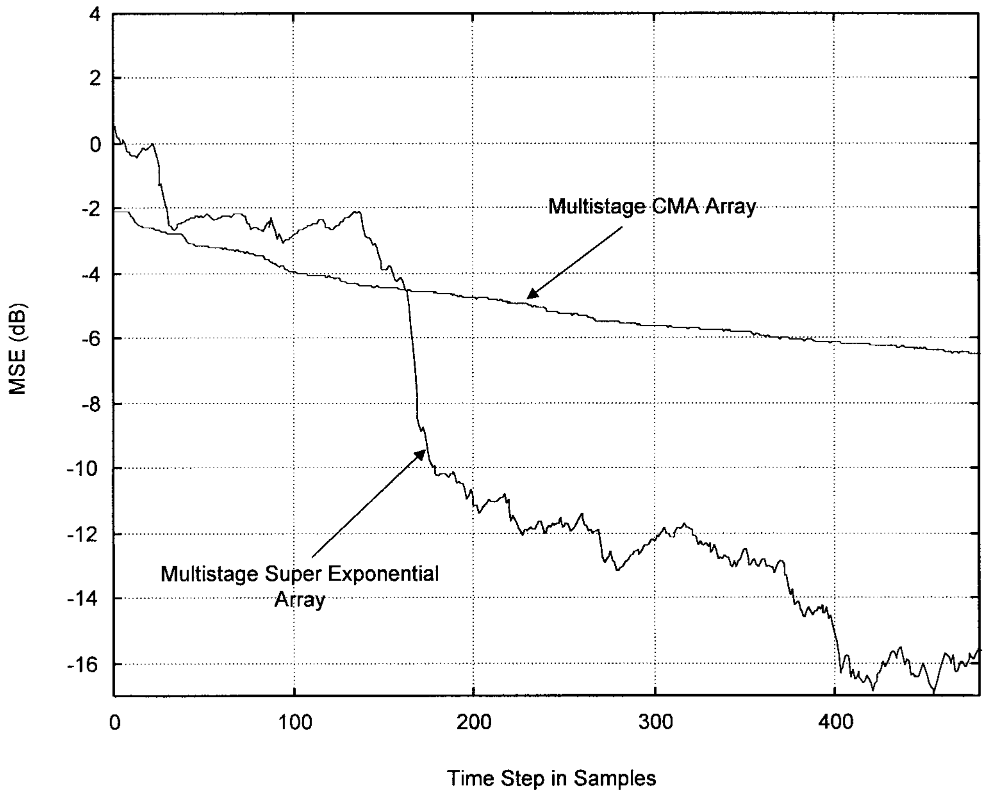

Fig. 5. Comparison with the CMA array. The MSE in decibels is relative to the first stage.

not necessarily imply slow convergence and, hence, extremely large sample size.

APPENDIX

A. The Super-Exponential Algorithm and Its Convergence

The iterative procedure [see (8) and (9)] applied to the vector $\tilde{\boldsymbol{s}}_{i_m}$ maintains the index of the tap with largest magnitude which was called in [3] the *leading tap* (for details, see [3, Section III]). It is globally convergent to $\tilde{\boldsymbol{\delta}}_{i_m}^{(m)}$ for any response vector $\tilde{\boldsymbol{s}}_{i_m}$ (for a proof, see [3]). An important aspect is the global convergence of the algorithm in the $\tilde{\boldsymbol{w}}_{i_m}^{(m)}$ domain to the same solution of the algorithm in the $\tilde{\boldsymbol{s}}_{i_m}^{(m)}$ domain: this may not be generally guaranteed.

First, observe that the procedure [see (8) and (9)] is a gradient-based search to solve the maximization problem

$$\max_{\tilde{\boldsymbol{s}}_{i_m}^{(m)}} \boldsymbol{\Psi}(\tilde{\boldsymbol{s}}_{i_m}^{(m)}) \tag{31}$$

subject to the constraint $\|\tilde{\boldsymbol{s}}_{i_m}^{(m)}\|^2 = 1$. In fact, the two steps in (8) and (9) are equivalent to the gradient-based iteration

$$\tilde{\boldsymbol{s}}_{i_m}^{(m)} = \tilde{\boldsymbol{s}}_{i_m}^{(m)} + \mu \frac{\partial \boldsymbol{\Psi}(\tilde{\boldsymbol{s}}_{i_m}^{(m)})}{\partial \tilde{\boldsymbol{s}}_{i_m}^{(m)}} \tag{32}$$

with $\boldsymbol{\Psi}(\tilde{\boldsymbol{s}}_{i_m}^{(m)}) = \Sigma_k \; s_{i_m}^{(m)}(k)^2 s_{i_m}(k)^{(m)^{*}2}, (\partial f(\boldsymbol{v})/\partial \boldsymbol{v})$ indicates the gradient[4] of the vector $\boldsymbol{v}$ and we choose a *very large* step size μ.

However, we have translated the maximization procedure for $\boldsymbol{\Psi}(\tilde{\boldsymbol{s}}_{i_m}^{(m)})$ over $\tilde{\boldsymbol{s}}_{i_m}^{(m)}$ into a maximization for a certain $\boldsymbol{\Psi}_{\tilde{\boldsymbol{w}}_{i_m}^{(m)}}(\tilde{\boldsymbol{w}}_{i_m}^{(m)})$ over $\tilde{\boldsymbol{w}}_{i_m}^{(m)}$ where

$$\boldsymbol{\Psi}_{\tilde{\boldsymbol{w}}_{i_m}^{(m)}}(\tilde{\boldsymbol{w}}_{i_m}^{(m)}) = \boldsymbol{\Psi}(\tilde{\boldsymbol{H}}_m \tilde{\boldsymbol{w}}_{i_m}^{(m)}) = \boldsymbol{\Psi}(\tilde{\boldsymbol{s}}_{i_m}^{(m)})$$

and we have used $\tilde{\boldsymbol{s}}_{i_m}^{(m)} = \tilde{\boldsymbol{H}}_m \tilde{\boldsymbol{w}}_{i_m}^{(m)}$.

Let us assume that $\tilde{\boldsymbol{s}}_{i_m}^{(m)E}$ is an extremum for $\boldsymbol{\Psi}(\tilde{\boldsymbol{s}}_{i_m}^{(m)})$, that is, $(\partial \boldsymbol{\Psi}(\tilde{\boldsymbol{s}}_{i_m}^{(m)})/\partial \tilde{\boldsymbol{s}}_{i_m}^{(m)})|_{\tilde{\boldsymbol{s}}_{i_m}^{(m)}=\tilde{\boldsymbol{s}}_{i_m}^{(m)E}} = \boldsymbol{0}$. It is then obviously true that $\tilde{\boldsymbol{w}}_{i_m}^{(m)E}$ such that $\tilde{\boldsymbol{s}}_{i_m}^{(m)E} = \tilde{\boldsymbol{H}}_m \tilde{\boldsymbol{w}}_{i_m}^{(m)E}$ is also an extremum for $\boldsymbol{\Psi}_{\tilde{\boldsymbol{w}}_{i_m}}(\tilde{\boldsymbol{w}}_{i_m})$ because $(\partial \boldsymbol{\Psi}_{\tilde{\boldsymbol{w}}_{i_m}}(\tilde{\boldsymbol{w}}_{i_m}^{(m)})/\partial \tilde{\boldsymbol{w}}_{i_m}^{(m)})|_{\tilde{\boldsymbol{w}}_{i_m}^{(m)}=\tilde{\boldsymbol{w}}_{i_m}^{(m)E}} = \boldsymbol{0}$.

The converse may not be true. That is, if we assume that $\overline{\tilde{\boldsymbol{w}}}_{i_m}^{(m)E}$ is an extremum for $\boldsymbol{\Psi}_{\tilde{\boldsymbol{w}}_{i_m}}(\tilde{\boldsymbol{w}}_{i_m})$, it may not be true that $\overline{\tilde{\boldsymbol{s}}}_{i_m}^{(m)E} = \tilde{\boldsymbol{H}}_m \overline{\tilde{\boldsymbol{w}}}_{i_m}^{(m)E}$ is an extremum for $\boldsymbol{\Psi}(\tilde{\boldsymbol{s}}_{i_m}^{(m)})$. In fact, we may have $(\partial \boldsymbol{\Psi}(\tilde{\boldsymbol{H}}_m \tilde{\boldsymbol{w}}_{i_m}^{(m)})/\partial \tilde{\boldsymbol{w}}_{i_m}^{(m)})|_{\tilde{\boldsymbol{w}}_{i_m}^{(m)}=\overline{\tilde{\boldsymbol{w}}}_{i_m}^{(m)E}} = \boldsymbol{0}$ if $(\partial \boldsymbol{\Psi}(\tilde{\boldsymbol{s}}_{i_m}^{(m)})/\partial \tilde{\boldsymbol{s}}_{i_m}^{(m)})|_{\overline{\tilde{\boldsymbol{s}}}_{i_m}^{(m)E}=\tilde{\boldsymbol{H}}_m \overline{\tilde{\boldsymbol{w}}}_{i_m}^{(m)E}} \neq \boldsymbol{0}$ belongs to the kernel of $\tilde{\boldsymbol{H}}_m$, which is orthogonal to the subspace spanned by $\tilde{\boldsymbol{H}}_m$, which means that $\overline{\tilde{\boldsymbol{w}}}_{i_m}^{(m)E}$ can be far from the desired solution.

[4]The derivative by a complex variable $u = u_r + ju_i$ is $(\partial/\partial u) = (\partial/\partial u_r) + j(\partial/\partial u_i)$, where $u_r = \mathrm{Real}[u]$ and $u_i = \mathrm{Imag}[u]$. Moreover, we also have $(\partial/\partial u^*) = (\partial/\partial u_r) - j(\partial/\partial u_i)$.

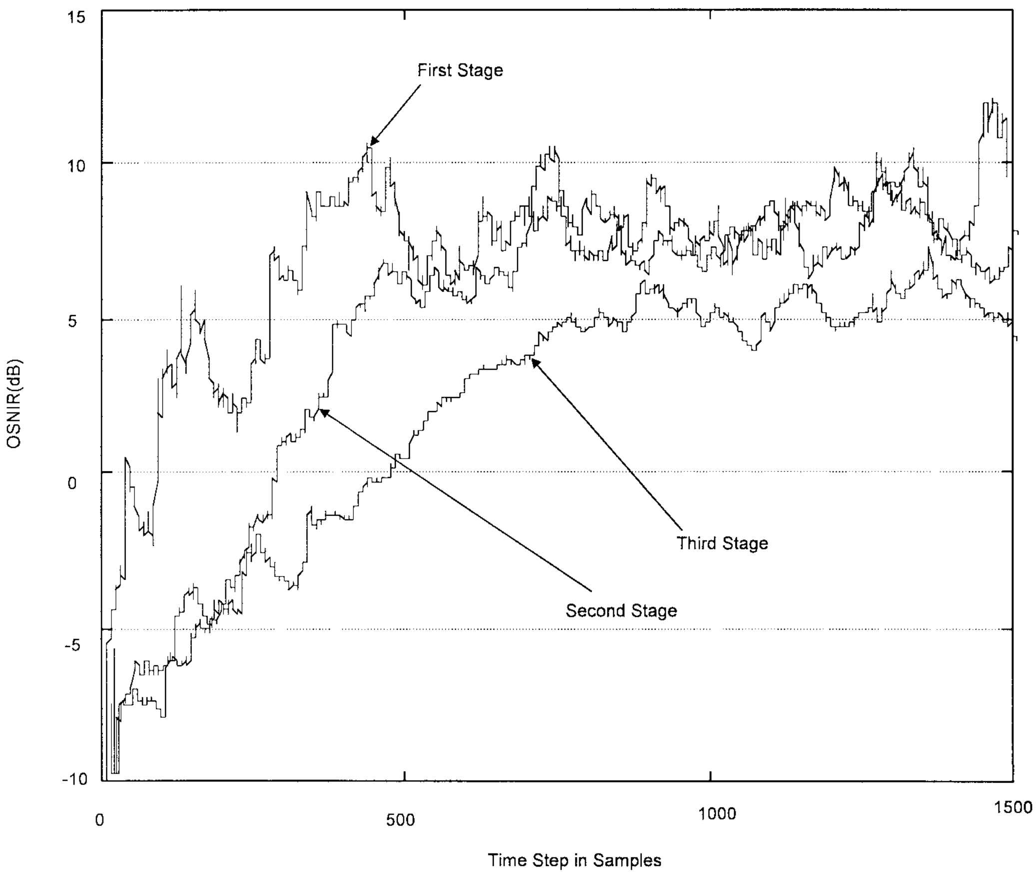

Fig. 6. Performance of the signal-to-noise-plus-interference ratio for the first three stages.

We investigate this last important issue. By the chain rule we have[5]

$$\frac{\partial \boldsymbol{\Psi}(\tilde{\boldsymbol{H}}_m \tilde{\boldsymbol{w}}_{i_m}^{(m)})}{\partial w_{i_m,n}^{(m)}} = \sum_{k=1}^{U_m} h_{n,k}^{(m)^*} \, \frac{\partial \boldsymbol{\Psi}(\tilde{\boldsymbol{s}}_{i_m}^{(m)})}{\partial s_{i_m,k}^{(m)}}\Bigg|_{s_{i_m,k}^{(m)}=\Sigma_{l=1}^{K} w_{i_m,l}^{(m)} h_{l,k}^{(m)}}$$

or in vector form

$$\frac{\partial \boldsymbol{\Psi}(\tilde{\boldsymbol{H}}_m \tilde{\boldsymbol{w}}_{i_m}^{(m)})}{\partial \tilde{\boldsymbol{w}}_{i_m}^{(m)}} = \tilde{\boldsymbol{H}}_m^H \frac{\partial \boldsymbol{\Psi}(\tilde{\boldsymbol{s}}_{i_m}^{(m)})}{\partial \tilde{\boldsymbol{s}}_{i_m}^{(m)}}\Bigg|_{\tilde{\boldsymbol{s}}_{i_m}^{(m)}=\tilde{\boldsymbol{H}}_m \tilde{\boldsymbol{w}}_{i_m}^{(m)}}. \tag{33}$$

Now multiply both sides of (33) by $\tilde{\boldsymbol{H}}_m(\tilde{\boldsymbol{H}}_m^H \tilde{\boldsymbol{H}}_m)^\dagger$ to obtain

$$\begin{aligned} \tilde{\boldsymbol{H}}_m(\tilde{\boldsymbol{H}}_m^H \tilde{\boldsymbol{H}}_m)^\dagger \frac{\partial \boldsymbol{\Psi}(\tilde{\boldsymbol{H}}_m \tilde{\boldsymbol{w}}_{i_m}^{(m)})}{\partial \tilde{\boldsymbol{w}}_{i_m}^{(m)}} &\simeq \boldsymbol{I}_{U_m} \frac{\partial \boldsymbol{\Psi}(\tilde{\boldsymbol{s}}_{i_m}^{(m)})}{\partial \tilde{\boldsymbol{s}}_{i_m}^{(m)}}\Bigg|_{\tilde{\boldsymbol{s}}_{i_m}^{(m)}=\tilde{\boldsymbol{H}}_m \tilde{\boldsymbol{w}}_{i_m}^{(m)}} \\ &= \frac{\partial \boldsymbol{\Psi}(\tilde{\boldsymbol{s}}_{i_m}^{(m)})}{\partial \tilde{\boldsymbol{s}}_{i_m}^{(m)}}\Bigg|_{\tilde{\boldsymbol{s}}_{i_m}^{(m)}=\tilde{\boldsymbol{H}}_m \tilde{\boldsymbol{w}}_{i_m}^{(m)}} \end{aligned} \tag{34}$$

because $\tilde{\boldsymbol{H}}_m(\tilde{\boldsymbol{H}}_m^H \tilde{\boldsymbol{H}}_m)^\dagger \tilde{\boldsymbol{H}}_m^H \simeq \boldsymbol{I}_{U_m}$, the $U_m \times U_m$ identity matrix. It is then evident from (34) that if $\overline{\tilde{\boldsymbol{w}}}_{i_m}^{(m)E}$ is an extremum for $\boldsymbol{\Psi}_{\tilde{\boldsymbol{w}}_{i_m}}(\tilde{\boldsymbol{w}}_{i_m})$, that is, if $(\partial \boldsymbol{\Psi}(\tilde{\boldsymbol{H}}_m \tilde{\boldsymbol{w}}_{i_m}^{(m)})/\partial \tilde{\boldsymbol{w}}_{i_m}^{(m)})|_{\tilde{\boldsymbol{w}}_{i_m}^{(m)}=\overline{\tilde{\boldsymbol{w}}}_{i_m}^{(m)E}} = \boldsymbol{0}$ then $(\partial \boldsymbol{\Psi}(\tilde{\boldsymbol{s}}_{i_m}^{(m)})/\partial \tilde{\boldsymbol{s}}_{i_m}^{(m)}\Big|_{\overline{\tilde{\boldsymbol{s}}}_{i_m}^{(m)E}=\tilde{\boldsymbol{H}}_m \overline{\tilde{\boldsymbol{w}}}_{i_m}^{(m)E}} \simeq \boldsymbol{0}$. Since $\|(\partial \boldsymbol{\Psi}(\tilde{\boldsymbol{s}}_{i_m}^{(m)})/\partial \tilde{\boldsymbol{s}}_{i_m}^{(m)}\|$ is continuous with respect to $w_{i_m,n}^{(m)}$ and $h_{k,n}^{(m)}$, then we only need to require that for a sufficiently small $\Delta > 0$ it is satisfied

$$\left\| \tilde{\boldsymbol{H}}_m(\tilde{\boldsymbol{H}}_m^H \tilde{\boldsymbol{H}}_m)^\dagger \tilde{\boldsymbol{H}}_m^H - \boldsymbol{I}_{U_m} \right\|_F < \Delta \tag{35}$$

to guarantee that $\|(\partial \boldsymbol{\Psi}(\tilde{\boldsymbol{s}}_{i_m}^{(m)})/\partial \tilde{\boldsymbol{s}}_{i_m}^{(m)})\|$ is arbitrarily small which implies that there exist an extremum $\tilde{\boldsymbol{s}}_{i_m}^{(m)E}$ such that $\|\overline{\tilde{\boldsymbol{s}}}_{i_m}^{(m)E} - \tilde{\boldsymbol{s}}_{i_m}^{(m)E})\|$ is arbitrarily small. In other words the extremum for $\boldsymbol{\Psi}_{\tilde{\boldsymbol{w}}_{i_m}}(\tilde{\boldsymbol{w}}_{i_m})$ is arbitrarily close to the extremum for $\boldsymbol{\Psi}(\tilde{\boldsymbol{s}}_{i_m}^{(m)})$ if there exists a sufficiently small Δ such that (35) is verified.

[5]Observe that for complex random variables $\xi = (1/2)(\xi_r - j\xi_i)$ where $\xi_r = \mathrm{Real}[\xi]$ and $\xi_i = \mathrm{Imag}[\xi]$ are two real random variables.

B. Improving the Numerical Stability of the Algorithm

The structure of (26) reveals its similarity with a standard solution of a multichannel recursive least squares estimation (for the ith channel), when the *desired process* ([10]) is substituted by the process $\tilde{z}_{i_m}(k)$. From the derivation, it clearly follows that at each $k+1$ stage we wish to solve

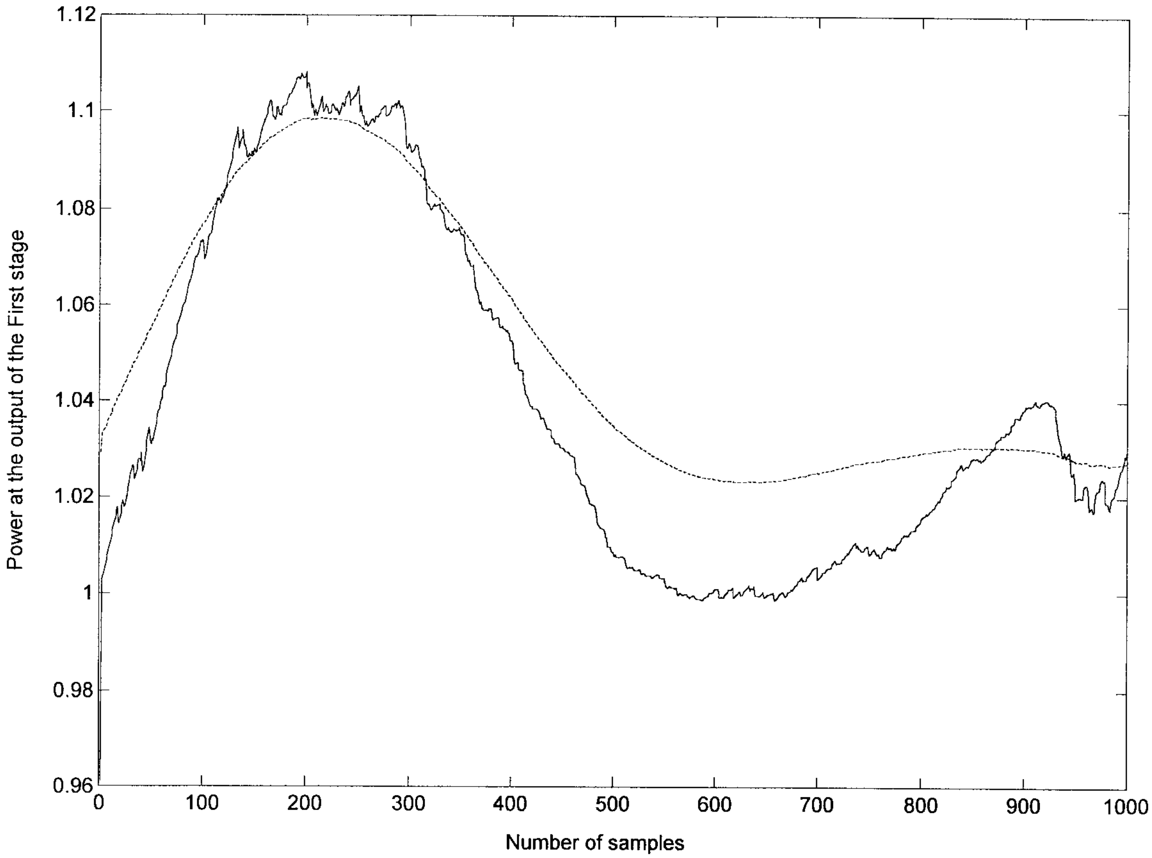

Fig. 7. Tracking performances of the QR approach. The channel is varying and the product maximum Doppler frequency-sample period is equal to 0.0006. The forgetting factor is equal to 0.97. The solid curve is the trace of the power for the first-stage output using the adaptive algorithm, and the dashed curve is the optimum solution variation.

the problem

$$\min_{\tilde{\boldsymbol{w}}_{i_m}^{(m)}} \left\| \begin{bmatrix} \lambda\tilde{\boldsymbol{Y}}^{(m)}(k) \\ \tilde{\boldsymbol{y}}^{(m)^T}(k+1) \end{bmatrix} \tilde{\boldsymbol{w}}_{i_m} - \begin{bmatrix} \lambda \boldsymbol{Z}_{i_m}^{(m)}(k) \\ \tilde{z}_{i_m}^{(m)}(k+1) \end{bmatrix} \right\|^2 \tag{36}$$

with $[\tilde{\boldsymbol{Y}}^{(m)}(k)]^T = [\tilde{\boldsymbol{y}}^{(m)}(1), \tilde{\boldsymbol{y}}^{(m)}(2), \cdots, \tilde{\boldsymbol{y}}^{(m)}(k)]$, $\boldsymbol{Z}_{i_m}^{(m)}(k) = [\tilde{z}_{i_m}^{(m)}(1), \tilde{z}_{i_m}^{(m)}(2), \cdots, \tilde{z}_{i_m}^{(m)}(k)]^T$. The *normal equations* define the desired minimizer as

$$\tilde{\boldsymbol{Y}}^{(m)^{T^*}}(k+1)\tilde{\boldsymbol{Y}}^{(m)}(k+1)\tilde{\boldsymbol{w}}_{i_m}^{(m)} = \tilde{\boldsymbol{Y}}^{(m)^{T^*}}(k+1)\boldsymbol{Z}_{i_m}^{(m)}(k+1).$$

Suppose that a matrix $\boldsymbol{V}(k)$ is known such that

$$\boldsymbol{Q}^T\tilde{\boldsymbol{Y}}^{(m)}(k) = \begin{bmatrix} \boldsymbol{V}(k) \\ 0 \end{bmatrix}$$

with $\boldsymbol{Q}$ orthogonal and $\boldsymbol{V}(k)$ being upper triangular matrix, then the problem stated in (36) is equivalent to

$$\min_{\tilde{\boldsymbol{w}}_{i_m}^{(m)}} \left\| \begin{bmatrix} \lambda \boldsymbol{V}(k) \\ \tilde{\boldsymbol{y}}(k+1)^T \end{bmatrix} \tilde{\boldsymbol{w}}_{i_m}^{(m)} - \begin{bmatrix} \lambda \overline{\boldsymbol{Z}}_{i_m}^{(m)}(k) \\ \tilde{z}_{i_m}(k+1) \end{bmatrix} \right\|^2 \tag{37}$$

where

$$\boldsymbol{Q}^T\boldsymbol{Z}_{i_m}^{(m)}(k) = \begin{bmatrix} \overline{\boldsymbol{Z}}_{i_m}^{(m)}(k) \\ \overline{z} \end{bmatrix}.$$

This equivalence can be seen by forming the normal equations for both problems and comparing them. The advantage is that the solution minimizer of (36) is simply the solution of a triangular system ([19]). This avoids the covariance matrix inversion and improves the performance of the recursion, when ill-conditioned data matrices are available. To find the matrix $\boldsymbol{Q}$, an efficient procedure can be adopted: a set of Givens rotations can be used to annihilate the lower triangular part of the matrix $\tilde{\boldsymbol{Y}}^{(m)}(k+1)$. The update is performed on the change in the parameter $\boldsymbol{w}_{i_m}^{(m)}(k)$ as $d\tilde{\boldsymbol{w}}_{i_m}^{(m)}(k) = \tilde{\boldsymbol{w}}_{i_m}^{(m)}(k+1) - \tilde{\boldsymbol{w}}_{i_m}^{(m)}(k)$ ([19]). The algorithm consists of the following steps.

1) Computation of the prediction error $u^{(m)}(k+1) = \tilde{z}_{i_m}^{(m)}(k+1) - \tilde{\boldsymbol{y}}^{(m)^T}(k+1)\tilde{\boldsymbol{w}}_{i_m}^{(m)}(k)$.
2) Form the matrix

$$\begin{bmatrix} \lambda \boldsymbol{V}(k) & \boldsymbol{0} \\ \tilde{\boldsymbol{y}}^{(m)^T}(k+1) & u^{(m)}(k+1) \end{bmatrix}.$$

3) Sweep the bottom part of this matrix using the Givens rotations.
4) Solve the triangular system $\boldsymbol{V}(k+1)d\tilde{\boldsymbol{w}}_{i_m}^{(m)}(k) = \overline{\boldsymbol{Z}}_{i_m}^{(m)}(k+1)$.
5) Obtain $\tilde{\boldsymbol{w}}_{i_m}^{(m)}(k+1) = \tilde{\boldsymbol{w}}_{i_m}^{(m)}(k) + d\tilde{\boldsymbol{w}}_{i_m}^{(m)}(k)$.

The computational complexity of the algorithm is only marginally increased with respect to the standard multichannel

RLS adaptive filter using the QR approach. The computations needed to update the process $\tilde{z}_{i_m}^{(m)}(k)$ are in fact absent in the standard square root RLS filter.

REFERENCES

[1] M. Martone, "Non-Gaussian multivariate adaptive AR estimation using the super exponential algorithm," *IEEE Trans. Signal Processing*, vol. 44, pp. 2640–2644, Oct. 1996.

[2] ——, "On-board regeneration of uplink signals using a blind adaptive multichannel estimator," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 34, pp. 49–62, Jan. 1998.

[3] O. Shalvi and E. Weinstein, "Super exponential methods for blind deconvolution," *IEEE Trans. Inform. Theory*, vol. 39, pp. 504–519, Mar. 1993.

[4] W. A. Gardner, *Statistical Spectral Analysis: A Nonprobabilistic Theory*. Englewood Cliffs, NJ: Prentice-Hall, 1987.

[5] O. Shalvi and E. Weinstein, "Universal methods for blind deconvolution," in *Blind Deconvolution*, S. Haykin, Ed. Englewood Cliffs, NJ: Prentice-Hall, 1994.

[6] J. F. Cardoso and A. Souloumiac, "Blind beamforming for non-Gaussian signals," *Proc. Inst. Elect. Eng.*, vol. 140, pt. F, pp. 362–370, Dec. 1993.

[7] M. C. Dogan and J. M. Mendel, "Cumulant-based blind optimum beamforming," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 30, pp. 722–740, July 1994.

[8] L. Tong, R. Liu, V. Soon, and Y. Huang, "Indeterminacy and identifiability of blind identification," *IEEE Trans. Circuits Syst.*, vol. 38, pp. 499–509, May 1991.

[9] S. Anderson, M. Millnert, M. Viberg, and B. Wahlberg, "An adaptive array for mobile communication systems," *IEEE Trans. Veh. Technol.*, vol. 40, Feb. 1991.

[10] S. Haykin, *Adaptive Filter Theory*. Englewood Cliffs, NJ: Prentice-Hall, 1996.

[11] J. M. Mendel, "Tutorial on higher order statistics (spectra) in signal processing and system theory: Theoretical results and some applications," *Proc. IEEE*, vol. 79, pp. 278–305, Mar. 1991.

[12] J. R. Treichler and B. G. Agee, "A new approach to multipath correction of constant modulus signals," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. 31, pp. 349–372, 1983.

[13] J. R. Treichler and M. G. Larimore, "New processing techniques based on the constant modulus adaptive algorithm," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. 33, pp. 420–431, 1985.

[14] J. Shynk, A. V. Keerthi, and A. Matur, "Steady state analysis of the multistage constant modulus array," *IEEE Trans. Signal Processing*, vol. 44, pp. 948–960, Apr. 1996.

[15] J. Shynk and R. P. Gooch, "The constant modulus array for cochannel signal copy and direction finding," *IEEE Trans. Signal Processing*, vol. 44, pp. 652–660, Mar. 1996.

[16] D. N. Godard, "Self-recovering equalization and carrier tracking in two dimensional data communications systems," *IEEE Trans. Commun.*, vol. 28, pp. 1867–1875, Nov. 1980.

[17] B. G. Agee, S. V. Schell, and W. A. Gardner, "Spectral self-coherence restoral: A new approach to blind adaptive signal extraction using antenna arrays," *Proc. IEEE*, vol. 78, pp. 753–767, Apr. 1990.

[18] L. Castedo and A. R. Figueiras-Vidal, "An adaptive beamforming technique based on cyclostationary signal properties," *IEEE Trans. Signal Processing*, vol. 43, pp. 1637–1650, July 1995.

[19] J. F. Bobrow and W. Murray, "An algorithm for RLS identification of parameters that vary quickly with time," *IEEE Trans. Automat. Contr.*, vol. 38, pp. 351–354, Feb. 1993.

[20] P. Lewis, "QR based algorithms for multichannel adaptive least squares lattice filters," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. 30, pp. 421–432, Mar. 1990.

[21] EIA/IS-20-A, "Recommended minimum standards for 800-MHz cellular land stations," May 1988.

[22] EIA/TIA-553, "Mobile station-land station compatibility specification," Sept. 1989.

[23] J. G. Proakis, *Digital Communications*. New York: McGraw-Hill, 1989.

[24] P. A. Bello. "Characterization of randomly time variant linear channels," *IEEE Trans. Commun. Syst. Technol.*, vol. 11, pp. 360–393, Dec. 1963.

Massimiliano (Max) Martone (M'93) was born in Rome, Italy, and received the Italian Doctor in Electronic Engineering degree in 1990 from the University of Rome "La Sapienza," Rome.

He was with the Italian Air Force from 1990 to 1991 and consulted in the area of DSP applied to communications for Staer, Inc., S.P.E., Inc., and TRS-Alfa Consult, Inc. In 1991, he joined the DSP staff at the On Board Equipment Division of Alenia Spazio, where he was involved in the design of DSP-based receivers and spread-spectrum transponders. In 1994, he was appointed as a Visiting Researcher at Rensselaer Polytechnic Institute, Troy, NY. He was a Wireless Communications Consultant for ATS, Inc., Waltham, MA, from 1994 to 1995. In 1995, he joined the Telecommunications Group, Watkins-Johnson Company, Gaithersburg, MD, where he currently leads the Advanced Wireless Development section in the design of equipment for AMPS, IS-136, GSM, and third-generation (3G) systems. His interests are in advanced signal processing for digital radios implementation, spread-spectrum multiple-access communications, and mobile radio communications.

Dr. Martone is a member of the New York Academy of Sciences and American Association for the Advancement of Science.