Low-Complexity Constrained Affine-Projection Algorithms

Stefan Werner, *Member, IEEE*, José Antonio Apolinário Jr., *Senior Member, IEEE*, Marcello L. R. de Campos, *Senior Member, IEEE*, and Paulo S. R. Diniz, *Fellow, IEEE*

***Abstract*—This paper proposes low-complexity constrained affine-projection (CAP) algorithms. The algorithms are suitable for linearly constrained filtering problems often encountered in communications systems. The CAP algorithms derived in this paper trade convergence speed and computational complexity in the same way as the conventional affine-projection (AP) algorithm. In addition, data-selective versions of the CAP algorithm are derived based on the concept of set-membership filtering. The set-membership constrained affine-projection (SM-CAP) algorithms include several constraint sets in order to construct a space of feasible solutions for the coefficient updates. The SM-CAP algorithms include a data-dependent step size that provides fast convergence and low mean-squared error. The paper also discusses important aspects of convergence and stability of constrained normalized adaptation algorithms and shows that normalization may introduce bias in the final solution.**

***Index Terms*—Adaptive filtering, affine projection algorithms, antenna array, beamforming, set-membership filtering, smart antennas.**

I. Introduction

Linearly constrained adaptive filters (LCAFs) find applications in several areas of signal processing, e.g., beamforming, blind interference suppression in code-division multiple-access (CDMA) systems, and system identification. The linear constraints usually reflect prior knowledge of certain parameters or properties of the problem under consideration, e.g., knowledge of direction of arrival (DOA) of user signals in antenna array processing [1], user spreading codes in blind multiuser detection [2], or linear phase feature of a plant in system identification [3].

Two alternative structures for implementation of the LCAF are the *direct-form* structure as in [1] and the *generalized sidelobe canceller* (GSC) structure of [4], [5]. Adaptive implementations of the direct-form structure incorporate the linear constraints into the adaptation algorithm in order to solve explicitly a constrained optimization problem. The GSC structure solves the same optimization problem as the direct-form structure filter by splitting the filter coefficient vector into two components operating on orthogonal subspaces. An advantage of adaptation algorithms derived for a direct-form structure is a potentially lower computational complexity than the adaptive GSC structure [6]. On the other hand, the GSC structure offers the advantage of using several unconstrained adaptation algorithms. A framework combining the advantages of the direct-form and the GSC structures was proposed in [7] and [8] such that unconstrained adaptation algorithms could be applied to a constrained problem while keeping the computational complexity similar to that of the direct-form structure.

The adaptation algorithms for linearly constrained problems proposed in the literature can be loosely categorized as least-mean-square (LMS)-type or recursive least-squares (RLS)-type algorithms (see, e.g., [1], [3], [6], and [9]). The constrained LMS (CLMS) algorithm [1] is attractive due to its low computational complexity; however, it suffers from slow convergence speed for correlated input signals. The more complex constrained RLS (CRLS) [9] algorithm has fast convergence but may be unstable even for well-behaved input signals. Similarly to the case of the conventional LMS and RLS algorithms, the CLMS and CRLS algorithms represent two extremes in terms of complexity and convergence speed.

The goal of this paper is to derive linearly constrained adaptive filtering algorithms with computational complexity and convergence speed between those of the CLMS and CRLS algorithms. We approach this problem using two techniques: 1) *data reusing* or projection onto affine subspaces and 2)*set-membership filtering* (SMF).

The concept of data reusing for conventional adaptive filters was introduced with the conventional affine-projection (AP) algorithm [10]–[12]. By adjusting the number of projections, or alternatively, the number of reuses, the AP algorithm can obtain ramping performances from that of the normalized LMS (NLMS) algorithm to that of the sliding-window RLS algorithm [13], [14].

SMF [15]–[21] is a recent approach to reduce computational complexity in adaptive filtering. SMF algorithms employ a deterministic objective function related to a bounded error constraint on the filter output such that the updates belong to a set of feasible solutions. The SMF algorithms feature reduced computational complexity primarily due to *data-selective* updates rendering an overall complexity that is usually much less than that of their conventional counterparts. The sparse updating in time can provide substantial savings in computations because it enables sharing of processor capacity [17] and less power consumption. A linearly constrained SMF algorithm of LMS-

Manuscript received April 16, 2004; revised February 9, 2005. The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Fulvio Gini.

S. Werner is with the Helsinki University of Technology, FIN-02015 HUT, FINLAND (e-mail: stefan.werner@hut.fi).

J. A. Apolinário Jr., is with the Departamento de Engenharia Elétrica, Instituto Militar de Engenharia, Rio de Janeiro, RJ 22290-270, Brazil (e-mail: apolin@ieee.org).

M. L. R. de Campos is with the COPPE/Universidade Federal do Rio de, Janeiro, Rio de Janeiro, RJ 21945-970, Brazil (e-mail: campos@lps.ufrj.br).

P. S. R. Diniz is with the COPPE/Universidade Federal do Rio de Janeiro, Rio de Janeiro, RJ 21945-970, Brazil (e-mail: diniz@lps.ufrj.br).

Digital Object Identifier 10.1109/TSP.2005.859348

type for a single constraint was proposed in [22]. Although belonging to another category of algorithms, the interior-point least-squares (IPLS) algorithm [23] is an interesting alternative approach also working with feasibility sets.

This paper proposes and analyzes a constrained affine-projection (CAP) algorithm using the same framework already used for other normalized constrained algorithms, such as the normalized constrained LMS (NCLMS) and binormalized data-reusing constrained LMS (BNDR-CLMS) algorithms [24]. The ideas of normalized constrained algorithms are extended to the framework of SMF [16], from which SM-CAP algorithms are derived. The SM-CAP algorithms, which can also be seen as a constrained version of the set-membership affine-projection (SM-AP) algorithm [20], retain the fast convergence of the CAP algorithm, and low misadjustment is obtained due to the data-selective property. The *a posteriori* output constrained LMS (APOC-LMS) algorithm proposed in [22] bears similarity to the proposed SM-CAP algorithm for the special case of one data reuse and a single constraint. However, even for this particular choice of parameters, our approach differs from that in [22] by the use of a correction term that prevents accumulation of errors when implemented in finite precision. Finally, the convergence analysis is provided for the CAP algorithm.

The paper is organized as follows. Section II presents the derivation of the CAP algorithm. Section III briefly reviews the basic concepts of SMF and introduces the SM-CAP algorithms. Computational complexity and convergence issues are addressed in Section IV. Simulations of the algorithms are shown in Section V, and conclusions are summarized in Section VI.

II. Constrained Affine-Projection Algorithm

The goal of this section is to derive an affine-projection algorithm for solving linearly constrained filtering problems. The CAP algorithm developed below can vary the number of data reuses to find an acceptable tradeoff between computational complexity and convergence speed.

In linearly constrained adaptive filtering, the constraints are given by the following set of J equations:

$$\mathbf{C}^T\mathbf{w} = \mathbf{f} \tag{1}$$

where $\mathbf{C}$ is an $N \times J$ constraint matrix and $\mathbf{f}$ is a vector containing the J constraint values.

The CAP algorithm to be derived solves the following optimization problem:

$$\begin{aligned} \mathbf{w}(k+1) = \arg\min_{\mathbf{w}} \|\mathbf{w} - \mathbf{w}(k)\|^2 \text{ subject to} \\ \mathbf{C}^T\mathbf{w} = \mathbf{f} \\ \mathbf{d}(k) - \mathbf{X}^T(k)\mathbf{w} = \mathbf{0} \end{aligned} \tag{2}$$

where $\mathbf{d}(k) \in \mathbb{R}^{L\times 1}$ and $\mathbf{X}(k) \in \mathbb{R}^{N\times L}$ are the desired-signal vector and input-signal matrix, defined by

$$\begin{aligned} \mathbf{d}(k) &= [d(k)\, d(k-1)\, \cdots\, d(k-L+1)]^T \\ \mathbf{X}(k) &= [\mathbf{x}(k)\, \mathbf{x}(k-1)\, \cdots\, \mathbf{x}(k-L+1)] \\ \mathbf{x}(k) &= [x(k)\, x(k-1)\, \cdots\, x(k-N+1)]^T \end{aligned} \tag{3}$$

and $\mathbf{w}(k) \in \mathbb{R}^{N\times 1}$ is the coefficient vector at time instant k.

TABLE I
Constrained Affine-Projection Algorithm

CAP Algorithm
$\mathbf{w}(0) = \mathbf{F}$
for each k
{
$\mathbf{X}(k) = [\mathbf{x}(k)\, \mathbf{x}(k-1)\, \cdots\, \mathbf{x}(k-L+1)]$
$\mathbf{e}(k) = \mathbf{d}(k) - \mathbf{X}^{\mathrm{T}}(k)\mathbf{w}(k)$
$\mathbf{t}(k) = \left[\mathbf{X}^{\mathrm{T}}(k)\, \mathbf{P}\, \mathbf{X}(k) + \delta\mathbf{I}\right]^{-1} \mathbf{e}(k)$
$\mathbf{w}(k+1) = \mathbf{P}\left[\mathbf{w}(k) + \mu\mathbf{X}(k)\, \mathbf{t}(k)\right] + \mathbf{F}$
}

Using the method of Lagrange multipliers to solve (2), the CAP algorithm becomes [25]

$$\mathbf{w}(k+1) = \mathbf{P}[\mathbf{w}(k) + \mathbf{X}(k)\, \mathbf{t}(k)] + \mathbf{F} \tag{4}$$

with

$$\mathbf{t}(k) = [\mathbf{X}^T(k)\, \mathbf{P}\, \mathbf{X}(k)]^{-1}\mathbf{e}(k) \tag{5}$$

and

$$\mathbf{e}(k) = \mathbf{d}(k) - \mathbf{X}^T(k)\mathbf{w}(k). \tag{6}$$

Matrix $\mathbf{P}$ performs a projection onto the homogeneous hyperplane defined by $\mathbf{C}^T\mathbf{w} = \mathbf{0}$, and vector $\mathbf{F}$ moves the projected solution back to the constraint hyperplane, as given below:

$$\mathbf{P} = \mathbf{I} - \mathbf{C}(\mathbf{C}^{\mathrm{T}}\mathbf{C})^{-1}\mathbf{C}^T \tag{7}$$

$$\mathbf{F} = \mathbf{C}(\mathbf{C}^T\mathbf{C})^{-1}\mathbf{f}. \tag{8}$$

For the special case of $L = 1$ or $L = 2$, the above recursions are identical to those of the NCLMS and BNDR-CLMS algorithms [24], respectively. For the constrained algorithms mentioned here, the simplification of the correction term $\mathbf{P}\mathbf{w}(k) + \mathbf{F}$ to $\mathbf{w}(k)$ should be avoided, since accumulation of roundoff errors may cause the solution to drift away from the constraint hyperplane (for more details, see [1]). The equations of the CAP algorithm are summarized in Table I, where a step size $0 < \mu < 2$ (usually chosen between 0 and 1, see Section IV) is used. Note that, in order to improve robustness, a diagonal matrix $\delta\mathbf{I}$ (δ is a small constant) is employed to regularize the matrix to be inverted.

III. Set-Membership Constrained Affine-Projection Algorithm

This section reviews the basic concepts of SMF and proposes two algorithms whose updates belong to a set of feasible solutions spanned by L past time instants while also satisfying a set of linear constraints. The first algorithm, the SM-CAP [26], is a direct extension of the SM-AP algorithm to linearly constrained problems. The second algorithm, the set-membership reduced peak-complexity CAP (SM-REDCAP) algorithm, is derived in an attempt to reduce the peak complexity of the SM-CAP algorithm. A basic feature of both algorithms is that if $\mathbf{w}(k)$ is not in the solution set, coefficient updates are required. On the other hand, if $\mathbf{w}(k)$ already is in the solution set, no coefficient updates are required, resulting in $\mathbf{w}(k+1)=\mathbf{w}(k)$.

A. Set-Membership Filtering

SMF is a framework applicable to filtering problems. For a more detailed introduction to the concept of SMF, the reader is referred to [15]–[17]. Specification on the filter parameters is achieved by constraining the output estimation error to be smaller than a deterministic threshold. As a result of the bounded error constraint, there will exist a set of filters that satisfy the imposed condition rather than a single filter.

Let $\mathcal{H}(k)$ denote the set containing all vectors $\mathbf{w}$ for which the associated output error at time instant k is upper bounded in magnitude by γ. In other words

$$\mathcal{H}(k)=\{\mathbf{w}\in\mathbb{R}^{N\times 1}:|e(k)|\leq\gamma\} \tag{9}$$

where $e(k)=d(k)-\mathbf{w}^T\mathbf{x}(k)$. The set $\mathcal{H}(k)$ is referred to as the *constraint set*, and its boundaries are hyperplanes.

Finally, let us define the *exact membership set* $\psi(k)$ as the intersection of the constraint sets over the time instants $i=1,\ldots,k$, i.e.,

$$\psi(k)=\bigcap_{i=1}^{k}\mathcal{H}(i). \tag{10}$$

The idea of set-membership adaptive recursion techniques (SMART) is to find adaptively an estimate that belongs to the exact membership set by reusing one or several constraint sets. One approach is to apply one of the many optimal bounding ellipsoid (OBE) algorithms, e.g., [17] and [27], in an attempt to outer bound the exact membership set $\psi(k)$ with ellipsoids. Another adaptive approach is the computation of a point estimate through projections using, for example, the information provided by the constraint set $\mathcal{H}(k)$. This is the approach used for derivation of the set-membership NLMS (SM-NLMS) algorithm [16]. The SM-AP algorithm [20] uses the information provided by the L past constraint sets.

B. The SM-CAP Algorithm

Our objective here is the derivation of an algorithm whose coefficients belong to the hyperplane defined by the linear constraints $\mathbf{C}^T\mathbf{w}=\mathbf{f}$ *and also* to the L last constraint sets. For this formulation, we express the exact membership set in (10) as $\psi(k)=\psi(k-L)\cap\psi_L(k)$, where $\psi_L(k)$ corresponds to the intersection of the L last constraint sets

$$\psi_L(k)=\bigcap_{i=k-L+1}^{k}\mathcal{H}(i). \tag{11}$$

Next, we consider the derivation of a data-selective algorithm whose coefficients belong to the hyperplane defined by (1) and also to the last L constraint sets, i.e., $\mathbf{C}^T\mathbf{w}=\mathbf{f}$ and $\mathbf{w}\in\psi_L(k)$. Let us state the following optimization criterion whenever $\mathbf{w}(k)\notin\psi_L(k)$:

$$\begin{aligned}\mathbf{w}(k+1)=\arg\min_{\mathbf{w}}\|\mathbf{w}-\mathbf{w}(k)\|^2 \text{ subject to}\\ \mathbf{C}^T\mathbf{w}=\mathbf{f}\\ \mathbf{d}(k)-\mathbf{X}^T(k)\mathbf{w}=\mathbf{g}(k)\end{aligned} \tag{12}$$

where $\mathbf{g}(k)=[g_1(k)\,g_2(k)\ldots g_L(k)]^T$ specifies the point in $\psi_L(k)$ for the update. To be consistent with (11), the elements of $\mathbf{g}(k)$ should be chosen such that $|g_i(k)|\leq\gamma$ for $i=1\ldots L$.

The solution obtained by applying the method of Lagrange multipliers is given by (13), shown at the bottom of the page, where

$$\mathbf{e}(k)=[e(k)\,\epsilon(k-1)\ldots\epsilon(k-L+1)]^T \tag{14}$$

and $\epsilon(k-i)=d(k-i)-\mathbf{x}^T(k-i)\mathbf{w}(k)$ denoting the *a posteriori* error at iteration $k-i$, with $\mathbf{P}$ and $\mathbf{F}$ given by (7) and (8), respectively.

The SM-CAP version in [26] chooses $g_i(k)=\epsilon(k-i+1)$, for $i\neq 1$ such that all but the first element in the vector $\mathbf{e}(k)-\mathbf{g}(k)$ of (13) are canceled, and $g_1(k)=\gamma e(k)/|e(k)|$ such that the *a posteriori* error lies on the closest boundary of $\mathcal{H}(k)$, yielding the update recursion [26]

$$\mathbf{w}(k+1)=\mathbf{P}\Big[\mathbf{w}(k)+\mathbf{X}(k)\big[\mathbf{X}^T(k)\mathbf{P}\mathbf{X}(k)\big]^{-1}\alpha(k)e(k)\mathbf{u}_1\Big]+\mathbf{F} \tag{15}$$

where $\mathbf{u}_1=[1\,0\ldots 0]^T$ and

$$\alpha(k)=\begin{cases}1-\frac{\gamma}{|e(k)|} & \text{if } |e(k)|>\gamma\\ 0 & \text{otherwise}\end{cases} \tag{16}$$

is the data dependent step size.

The equations of the SM-CAP algorithm [26] are summarized in Table II, and a graphical description in $\mathbb{R}^2$ is shown in Fig. 1 for the case of $N=2$ and $L=1$. For the particular case of $L=1$, the SM-CAP algorithm is, apart from the correction term, identical to the APOC-LMS algorithm proposed in [22]. In our formulation, with the use of a correction term as pointed out in Section II, no accumulation of roundoff errors will cause the solution to drift away from the constraint hyperplane. The departure from the constraint plane is generally slower for the APOC-LMS algorithm as compared with the NCLMS without correction term due to the data-selective updating [21] coming from the SMF approach to adaptive filtering.

Remark 1: Whenever an update is needed, part or all of the elements of matrix $\mathbf{X}^T(k)\mathbf{P}\mathbf{X}(k)$ need to be recalculated, increasing the computational burden per update, especially, close to a steady-state solution, when updating is sparse in time. However, this may not pose any major problem since the reduced frequency of updating more than compensates for the increase in complexity introduced by $\mathbf{X}^T(k)\mathbf{P}\mathbf{X}(k)$.

$$\mathbf{w}(k+1)=\begin{cases}\mathbf{P}\left\{\mathbf{w}(k)+\mathbf{X}(k)[\mathbf{X}^T(k)\mathbf{P}\mathbf{X}(k)]^{-1}[\mathbf{e}(k)-\mathbf{g}(k)]\right\}+\mathbf{F} & \text{if } |e(k)|>\gamma\\ \mathbf{w}(k) & \text{otherwise}\end{cases} \tag{13}$$

TABLE II
SET-MEMBERSHIP CONSTRAINED AFFINE-PROJECTION ALGORITHM

SM-CAP Algorithm

$\mathbf{w}(0) = \mathbf{F}$
for each k
{
  $e(k) = d(k) - \mathbf{x}^{\mathrm{T}}(k)\mathbf{w}(k)$
  if $|e(k)| > \gamma$
    $\alpha(k) = 1 - \gamma/|e(k)|$
    $\mathbf{X}(k) = [\mathbf{x}(k)\,\mathbf{x}(k-1)\,\cdots\,\mathbf{x}(k-L+1)]$
    $\mathbf{t}(k) = \left[\mathbf{X}^{\mathrm{T}}(k)\,\mathbf{P}\,\mathbf{X}(k) + \delta\mathbf{I}\right]^{-1}\alpha(k)e(k)\mathbf{u}_1$
    $\mathbf{w}(k+1) = \mathbf{P}\left[\mathbf{w}(k) + \mathbf{X}(k)\,\mathbf{t}(k)\right] + \mathbf{F}$
  else
    $\mathbf{w}(k+1) = \mathbf{w}(k)$
}

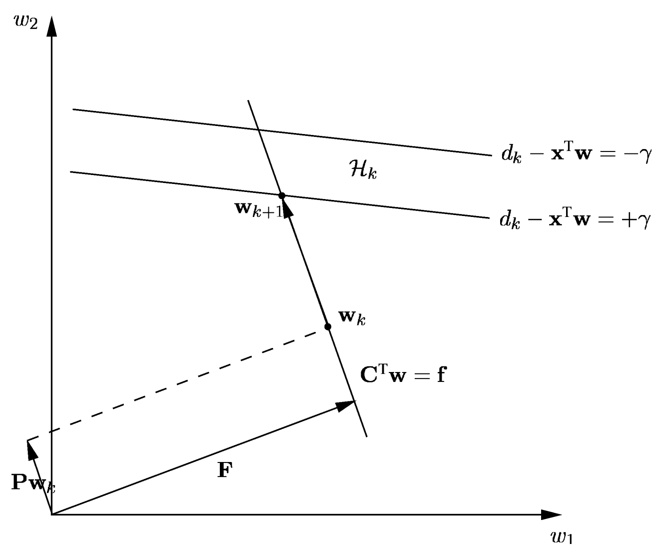

Fig. 1. Geometrical interpretation of the SM-CAP algorithm in $\mathbb{R}^2$ for $N = 2$ and $L = 1$.

In the following subsection, we consider an algorithm that can reuse all past calculations of the cross correlations in $\mathbf{X}^T(k)\mathbf{P}\mathbf{X}(k)$, even when updating is sparse in time.

C. The SM-REDCAP Algorithm

In an attempt to reduce the increased peak complexity of the SM-CAP algorithm for the case of nonconsecutive updating instants, we propose an alternative algorithm—the SM-REDCAP—which seeks a solution that belongs to *the L past constraint sets that contributed to an update*. This philosophy is in agreement with the underlying principles of quasi-OBE algorithms (see, e.g., [28]).

Let $\bar{\psi}_L(k)$ denote the member of the exact membership set $\psi(k)$ spanned by L arbitrary constraint sets, i.e.,

$$\bar{\psi}_L(k) = \bigcap_{i \in \mathcal{I}_L(k)} \mathcal{H}(i) \tag{17}$$

where $\mathcal{I}_L(k)$ is an index set with $\{i_j[k]\}_{j=1}^{L}$ taken from the set $\{1, \ldots, k\}$ specifying the time instants of constraint sets used in the update. This more general formulation includes the SM-CAP algorithm described in the previous section as a special case. For the SM-REDCAP algorithm considered here, $\mathcal{I}_L(k)$ simply reduces to

$$\mathcal{I}_L(k) = \begin{cases} \{k, i_1[k-1], i_2[k-1], \ldots i_{L-1}[k-1]\} & \text{if } |e(k)| > \gamma \\ \mathcal{I}_L(k-1) & \text{otherwise.} \end{cases} \tag{18}$$

Notice that $\mathcal{I}_L(k)$ in (18) is changing only if $|e(k)| > \gamma$, which is equivalent to verifying whether $\mathbf{w}(k) \in \mathcal{H}(k)$. This is because $\bar{\psi}_L(k)$ reuses $L-1$ constraint sets from $\bar{\psi}_L(k-1)$. Using a similar optimization criterion as in (12) and the same reasoning for choosing vector $\mathbf{g}(k)$, the updating recursions of the SM-REDCAP become (19), shown at the bottom of the page, where $\alpha(k)$ and $\mathbf{u}_1$ are the same as used with the SM-CAP algorithm. The input-signal matrix $\tilde{\mathbf{X}}(k)$ is given by

$$\tilde{\mathbf{X}}(k) = [\mathbf{x}(k)\,\mathbf{x}(i_2[k])\,\cdots\,\mathbf{x}(i_L[k])]. \tag{20}$$

Note that the input-signal matrix of the SM-REDCAP algorithm in (20) differs from that of the SM-CAP and CAP algorithms in (3) whenever there is a gap in successive updates (and only equal if L successive updates occur). The equations of the SM-REDCAP algorithm are presented in Table III.

Remark 2: If no update occurs for consecutive iterations, only the $L-1$ inner products $\{\mathbf{x}^T(k)\mathbf{P}\mathbf{x}(i_j[k])\}_{j=1}^{L-1}$ appearing in $\tilde{\mathbf{X}}^T(k)\mathbf{P}\tilde{\mathbf{X}}(k)$ need to be recalculated. As a consequence, the peak complexity of the SM-REDCAP algorithm will be lower than that of the SM-CAP algorithm as will be illustrated in the following section.

IV. COMPUTATIONAL COMPLEXITY AND CONVERGENCE ISSUES

This section deals initially with the computational complexity of the three algorithms—CAP, SM-CAP, and SM-REDCAP—in terms of number of multiplications and divisions. The analysis for the coefficient-error vector is addressed for two different scenarios: with and without reference or training signals. Section IV-B analyzes a system identification problem where a training sequence is available. An example of such setup is given in [3], where the plant is constrained to having linear phase. Thereafter, in Section IV-C, a system is analyzed where no training sequence is available.

$$\mathbf{w}(k+1) = \begin{cases} \mathbf{P}\left[\mathbf{w}(k) + \tilde{\mathbf{X}}(k)\left[\tilde{\mathbf{X}}^T(k)\mathbf{P}\tilde{\mathbf{X}}(k)\right]^{-1}\alpha(k)e(k)\mathbf{u}_1\right] + \mathbf{F} & \text{if } |e(k)| > \gamma \\ \mathbf{w}(k) & \text{otherwise} \end{cases} \tag{19}$$

TABLE III
SET-MEMBERSHIP REDUCED PEAK-COMPLEXITY CONSTRAINED AFFINE-PROJECTION ALGORITHM

SM-REDCAP Algorithm
$\mathbf{w}(0) = \mathbf{F}$
for each k
{
$e(k) = d(k) - \mathbf{x}^{\mathrm{T}}(k)\mathbf{w}(k)$
if $\lvert e(k)\rvert > \gamma$
$\alpha(k) = 1 - \gamma/\lvert e(k)\rvert$
$\tilde{\mathbf{X}} = [\mathbf{x}(k)\ \tilde{\mathbf{X}}(:, 1 : L-1)]$
$\mathbf{t}(k) = \left[\tilde{\mathbf{X}}^{\mathrm{T}}\,\mathbf{P}\,\tilde{\mathbf{X}} + \delta\mathbf{I}\right]^{-1} \alpha(k)e(k)\mathbf{u}_1$
$\mathbf{w}(k+1) = \mathbf{P}\left[\mathbf{w}(k) + \tilde{\mathbf{X}}\,\mathbf{t}(k)\right] + \mathbf{F}$
else
$\mathbf{w}(k+1) = \mathbf{w}(k)$
}

Examples of applications where no training is available are beamforming and multiuser detection [1], [2].

A. Computational Complexity

Computational complexity in terms of number of multiplications and divisions per update for the CAP, SM-CAP, and SM-REDCAP algorithms are shown in Table IV.

In the case of the CAP algorithm, computation of $\mathbf{X}^T(k)\mathbf{P}\mathbf{X}(k)$ requires calculation of the elements of its first row (or column) at every iteration (all other information is available from previous iterations). In the SM-REDCAP algorithm, the computation of $\mathbf{X}^T(k)\mathbf{P}\mathbf{X}(k)$ requires the calculation of its first row (or column) *and only when updating is necessary*. In the case of the SM-CAP algorithm, the computation of $\mathbf{X}^T(k)\mathbf{P}\mathbf{X}(k)$ requires the calculation of rows (or columns) from 1 to L, depending on the gap (in number of iterations) between successive updates.

In Table IV, for the SM-CAP algorithm, the maximum complexity is listed assuming that the coefficient vector is updated and also the worst case of computing $\mathbf{X}^T(k)\mathbf{P}\mathbf{X}(k)$ without any previously computed element. However, whenever updates occur within L iterations, previous information can be used to reduce number of operations.

The results in Table IV do not reflect one of the important gains of applying SMF algorithms, viz., the reduced number of required updates. For time instants where no updates are required, the complexity of the SMF algorithms is due to filtering only, i.e., $N-1$ additions and N multiplications.

Finally, note that the bound γ is a specification on the output error of the adaptive filter. For increasing γ, updates will occur less frequently. In certain applications, a slight increase in the mean-square error (MSE) is expected, as pointed out in [27]. In the case of system identification, the contrary may happen for a region close to the minimum MSE, as pointed out in [19].

TABLE IV
COMPUTATIONAL COMPLEXITY PER UPDATE: N IS THE NUMBER OF COEFFICIENTS IN $\mathbf{w}$, J IS THE NUMBER OF CONSTRAINTS, L IS THE NUMBER OF REUSED HYPERPLANES, AND K_{inv} IS A CONSTANT ASSOCIATED WITH THE COMPLEXITY OF THE METHOD USED TO IMPLEMENT THE MATRIX INVERSION REQUIRED IN (5)

ALG.	MULTIPLICATIONS	DIV.
CAP	$(3J+3L+1)N + (K_{inv}+1)L^2$	0
SM-CAP	$\left[2J + 2JL + L + \frac{L(L+1)}{2} + 1\right] N + K_{inv}L^2 + L + 1$	1
SM-REDCAP	$(4J+2L+1)N + K_{inv}L^2 + L + 1$	1

B. Analysis With Training Signal

The optimal solution $\mathbf{w}_{\mathrm{opt}}$ to the constrained optimization problem is given by [29], [30]

$$\mathbf{w}_{\mathrm{opt}} = \mathbf{R}^{-1}\mathbf{p} - \mathbf{R}^{-1}\mathbf{C}\left(\mathbf{C}^T\mathbf{R}^{-1}\mathbf{C}\right)^{-1}\left(\mathbf{C}^T\mathbf{R}^{-1}\mathbf{p} - \mathbf{f}\right) \tag{21}$$

where $\mathbf{R} = \mathrm{E}[\mathbf{x}(k)\mathbf{x}^T(k)]$ and $\mathbf{p} = \mathrm{E}[d(k)\mathbf{x}(k)]$ are the input-signal correlation matrix and cross-correlation vector, respectively.

Let $\mathbf{d}(k)$ be modeled as in a system identification setup, as follows:

$$\mathbf{d}(k) = \mathbf{X}^T(k)\mathbf{w}_{\mathrm{opt}}. \tag{22}$$

If the coefficient-error vector is defined as

$$\Delta\mathbf{w}(k) = \mathbf{w}(k) - \mathbf{w}_{\mathrm{opt}} \tag{23}$$

we obtain

$$\Delta\mathbf{w}(k+1) = \mathbf{P}\{\mathbf{I} - \mu\mathbf{X}(k)[\mathbf{X}^T(k)\mathbf{P}\mathbf{X}(k)]^{-1}\mathbf{X}^T(k)\}\Delta\mathbf{w}(k) + \mathbf{C}(\mathbf{C}^T\mathbf{C})^{-1}(\mathbf{f} - \mathbf{C}^T\mathbf{w}_{\mathrm{opt}}). \tag{24}$$

The constraints are clearly satisfied by the optimal solution, i.e., $\mathbf{f} - \mathbf{C}^T\mathbf{w}_{\mathrm{opt}} = \mathbf{0}$. As a result, the next expression follows:

$$\Delta\mathbf{w}(k+1) = \mathbf{P}\{\mathbf{I} - \mu\mathbf{X}(k)[\mathbf{X}^T(k)\mathbf{P}\mathbf{X}(k)]^{-1}\mathbf{X}^T(k)\}\Delta\mathbf{w}(k). \tag{25}$$

It can be verified that $\mathbf{P}\Delta\mathbf{w}(k) = \Delta\mathbf{w}(k)$, by noting that any coefficient vector satisfying the constraints $\mathbf{C}^T\mathbf{w} = \mathbf{f}$ can be decomposed as $\mathbf{w} = \mathbf{P}\mathbf{w} + \mathbf{F}$ such that

$$\begin{aligned}\mathbf{P}\Delta\mathbf{w}(k) &= \mathbf{P}\mathbf{w}(k) - \mathbf{P}\mathbf{w}_{\mathrm{opt}} = \mathbf{w}(k) - \mathbf{F} - \mathbf{P}\mathbf{w}_{\mathrm{opt}} \\ &= \mathbf{w}(k) - \mathbf{w}_{\mathrm{opt}} = \Delta\mathbf{w}(k).\end{aligned}$$

Also note that, due to the matrix inversion, (25) is valid only in cases where $(N-J) \geq L$.

Using the fact that matrix $\mathbf{P}$ is idempotent, i.e., $\mathbf{P}\mathbf{P} = \mathbf{P}$ and $\mathbf{P}^T = \mathbf{P}$, together with the relation $\mathbf{P}\Delta\mathbf{w}(k) = \Delta\mathbf{w}(k)$, we can write (25) as

$$\begin{aligned}\Delta\mathbf{w}(k+1) &= \mathbf{P}\{\mathbf{I} - \mu\mathbf{X}(k)[\mathbf{X}^T(k)\mathbf{P}\mathbf{X}(k)]^{-1}\mathbf{X}^T(k)\}\mathbf{P}\Delta\mathbf{w}(k) \\ &= \{\mathbf{I} - \mu\mathbf{P}\mathbf{X}(k)[\mathbf{X}^T(k)\mathbf{P}^T\mathbf{P}\mathbf{X}(k)]^{-1}\mathbf{X}^T(k)\mathbf{P}^T\}\Delta\mathbf{w}(k) \\ &= \{\mathbf{I} - \mu\bar{\mathbf{X}}(k)[\bar{\mathbf{X}}^T(k)\bar{\mathbf{X}}(k)]^{-1}\bar{\mathbf{X}}^T(k)\}\Delta\mathbf{w}(k) \\ &= \mathbf{T}(k)\Delta\mathbf{w}(k)\end{aligned} \tag{26}$$

where $\bar{\mathbf{X}}(k) = \mathbf{P}\mathbf{X}(k)$ and $\mathbf{T}(k) = \left\{\mathbf{I} - \mu\bar{\mathbf{X}}(k)[\bar{\mathbf{X}}^T(k)\bar{\mathbf{X}}(k)]^{-1}\bar{\mathbf{X}}^T(k)\right\}$.

We now examine the conditions for convergence with probability 1 (convergence everywhere) of the system describing $\Delta\mathbf{w}(k+1)$. In order to guarantee stability of the linear time-variant system of (26), consider the following lemma

Lemma 1: The CAP algorithm with the coefficient-error vector update given by (26) is stable, i.e., $\|\Delta\mathbf{w}(k+1)\| \leq \|\Delta\mathbf{w}(k)\|$, for $0 \leq \mu \leq 2$ and $N - J \geq L$.

Proof: Using the relation $\|\mathbf{AB}\| \leq \|\mathbf{A}\| \cdot \|\mathbf{B}\|$, we get

$$\begin{aligned}\|\Delta\mathbf{w}(k+1)\|_2 &= \|\mathbf{T}(k)\Delta\mathbf{w}(k)\|_2 \leq \|\mathbf{T}(k)\|_2 \cdot \|\Delta\mathbf{w}(k)\|_2 \\ &= \|\Delta\mathbf{w}(k)\|_2 \end{aligned} \quad (27)$$

where we used the fact that $\|\mathbf{T}(k)\|_2 = 1$ for $0 \leq \mu \leq 2$.

As for the asymptotic stability, we state the following theorem.

Theorem 1: If the input signal is persistently exciting and $N - J \geq L$, then the solution of (26) and, consequently, the CAP algorithm is asymptotically stable for $0 < \mu < 2$.

Proof: Using singular value decomposition (SVD), we can rewrite the transformed input matrix $\bar{\mathbf{X}}(k) = \mathbf{PX}(k)$ as $\bar{\mathbf{X}}(k) = \mathbf{U}(k)\Sigma(k)\mathbf{V}^T(k)$, where the unitary matrices $\mathbf{U}(k) \in \mathbb{R}^{N\times N}$ and $\mathbf{V}(k) \in \mathbb{R}^{L\times L}$ contain the left and right *singular vectors* respectively, and $\Sigma(k) \in \mathbb{R}^{N\times L}$ contains the *singular values* on its main diagonal. Consequently, we can write

$$\begin{aligned}\Delta\mathbf{w}(k+1) &= \mathbf{T}(k)\Delta\mathbf{w}(k) \\ &= \Big\{\mathbf{I} - \mu\mathbf{U}(k)\Sigma(k)\mathbf{V}^T(k)\left(\mathbf{V}(k)\Sigma^T(k)\Sigma(k)\mathbf{V}^T(k)\right)^{-1} \\ &\quad \times \mathbf{V}(k)\Sigma^T(k)\mathbf{U}^T(k)\Big\}\Delta\mathbf{w}(k) \\ &= \Big\{\mathbf{I} - \mu\mathbf{U}(k)\Sigma(k)\left(\Sigma^T(k)\Sigma(k)\right)^{-1}\Sigma^T(k)\mathbf{U}^T(k)\Big\}\Delta\mathbf{w}(k)\end{aligned} \quad (28)$$

where we used the fact that for two invertible matrices $\mathbf{A}$ and $\mathbf{B}$, $(\mathbf{AB})^{-1} = \mathbf{B}^{-1}\mathbf{A}^{-1}$.

Persistence of excitation of the input signal and condition $N - J \geq L$ guarantee that matrix $\left(\Sigma^T(k)\Sigma(k)\right)^{-1}$ exists. Matrix $\Sigma(k)\left(\Sigma^T(k)\Sigma(k)\right)^{-1}\Sigma^T(k) \in \mathbb{R}^{N\times N}$ is a diagonal matrix with L ones and $N - L$ zeros, i.e.,

$$\Sigma(k)(\Sigma^T(k)\Sigma(k))^{-1}\Sigma^T(k) = \begin{pmatrix}\mathbf{I}_{L\times L} & \mathbf{0}_{L\times(N-L)} \\ \mathbf{0}_{(N-L)\times L} & \mathbf{0}_{(N-L)\times(N-L)}\end{pmatrix}. \quad (29)$$

Therefore

$$\begin{aligned}\|\Delta\mathbf{w}(k+1)\|_2^2 &= \Delta\mathbf{w}^T(k)\mathbf{T}^T(k)\mathbf{T}(k)\Delta\mathbf{w}(k) \\ &= \Delta\mathbf{w}^T(k)\Big\{\mathbf{I} - \mu(2-\mu)\mathbf{U}(k)\Sigma(k) \\ &\quad \times \left(\Sigma^T(k)\Sigma(k)\right)^{-1}\Sigma^T(k)\mathbf{U}^T(k)\Big\}\Delta\mathbf{w}(k) \\ &= \|\Delta\mathbf{w}(k)\|_2^2 - \mu(2-\mu)\Delta\tilde{\mathbf{w}}^T(k)\Sigma(k) \\ &\quad \times \left(\Sigma^T(k)\Sigma(k)\right)^{-1}\Sigma^T(k)\Delta\tilde{\mathbf{w}}(k) \\ &= \|\Delta\mathbf{w}(k)\|_2^2 - \mu(2-\mu)\sum_{i=1}^{L}\Delta\tilde{w}^2(i)\end{aligned} \quad (30)$$

where $\Delta\tilde{\mathbf{w}}(k) = \mathbf{U}^T(k)\Delta\mathbf{w}(k)$.

For the asymptotic stability, we can conclude that $\|\Delta\mathbf{w}(k+1)\|_2^2$ remains constant, i.e., $\|\Delta\mathbf{w}(k+1)\|_2^2 = \|\Delta\mathbf{w}(k)\|_2^2$, during an interval $[k_1, k_2]$ if and only if we choose $\mu = 2$ or $\mu = 0$, or if $\Delta\mathbf{w}(k)$ is orthogonal to the L left singular vectors of $\bar{\mathbf{X}}(k) = \mathbf{PX}(k)$ corresponding to the L nonzero singular values in $\Sigma(k)$ for all $k \in [k_1, k_2]$, i.e.,

$$\mathbf{U}^T(k)\Delta\mathbf{w}(k) = [\underbrace{0,\ldots,0}_{L},\ \underbrace{*,\ldots,*}_{N-L}]^T, \qquad \forall k \in [k_1, k_2]$$

where the elements denoted $*$ can take arbitrary values. However, if the input signal is persistently exciting, we can define an infinite number of sets $\mathcal{S}_i = \{\bar{\mathbf{U}}(k_{1i}), \ldots, \bar{\mathbf{U}}(k_{2i})\}$, where $\bar{\mathbf{U}}(k) \in \mathbb{R}^{N\times L}$ denotes the L first columns of $\mathbf{U}(k)$, with $N' \leq (k_{2i} - k_{1i}) \leq N''$, such that each set $\mathcal{S}_i$ completely spans $\mathbb{R}^L$ for some finite value of $N'' > 0$. This makes it impossible to have $\Delta\mathbf{w}(k)$ orthogonal to all $\bar{\mathbf{U}}(k) \in \mathcal{S}_i$ and, as a consequence, $\|\Delta\mathbf{w}(k_{2i})\|_2^2 < \|\Delta\mathbf{w}(k_{1i})\|_2^2$. Since the number of sets is infinite, the coefficient-error norm is always reduced by the action of successive projections. Moreover, we know that $0 \leq \sum_{i=1}^{L}\Delta\tilde{w}^2(i) \leq \sum_{i=1}^{N}\Delta\tilde{w}^2(i) = \|\Delta\mathbf{w}(k)\|_2^2$, for $\mathbf{U}(k)\mathbf{U}^T(k) = \mathbf{I}$; then, we can write

$$\begin{aligned}\|\Delta\mathbf{w}(k+1)\|_2^2 &= \|\Delta\mathbf{w}(k)\|_2^2 - \mu(2-\mu)\kappa_k\|\Delta\mathbf{w}(k)\|_2^2 \\ &= \rho_k\|\Delta\mathbf{w}(k)\|_2^2\end{aligned} \quad (31)$$

where κ_k and ρ_k are scalars between 0 and 1, as far as μ is chosen properly. In order to establish that $\|\Delta\mathbf{w}(k)\|_2^2 \to 0$ for $k \to \infty$, we can run a convergence test on a series with all values of $\|\Delta\mathbf{w}(k)\|_2^2$ corresponding to $\rho_k < 1$. From the above, we know that there exists an infinite number of time instants where $\rho_k < 1$; then, we let $\mathcal{I}_{\rho_k<1}$ denote the infinite index set containing the time instants when $\rho_k < 1$. According to the *ratio test*, the series $\sum_{k\in\mathcal{I}_{\rho_k<1}}\|\Delta\mathbf{w}(k)\|_2^2$ converges, and, therefore, we must have $\|\Delta\mathbf{w}(k+1)\|_2^2 \to 0$ for $k \to \infty$, which concludes the proof. ■

Observation: Equation (30) indicates that practical step sizes are in the range $0 < \mu \leq 1$, since larger step sizes neither increase the speed of convergence nor further reduce the coefficient-error norm.

If observation noise is present in the reference-signal vector, i.e., $\mathbf{d}(k) = \mathbf{X}^T(k)\mathbf{w}_{\text{opt}} + \mathbf{n}(k)$, where $\mathbf{n}(k) = [n(k)\ n(k-1)\ \cdots\ n(k-L+1)]^T$, $\Delta\mathbf{w}(k+1)$ shall be modified to include an additional term $\boldsymbol{\xi}_n$, given by

$$\boldsymbol{\xi}_n = \mu\mathbf{PX}(k)[\mathbf{X}^T(k)\mathbf{PX}(k)]^{-1}\mathbf{n}(k). \quad (32)$$

Assuming noise to be uncorrelated and zero mean, this term averages out to zero and does not contribute to bias.

C. Analysis Without Training Signal

In the case where no training sequence is available, the optimum solution is given by [1]

$$\mathbf{w}_{\text{opt}} = \mathbf{R}^{-1}\mathbf{C}\left(\mathbf{C}^T\mathbf{R}^{-1}\mathbf{C}\right)^{-1}\mathbf{f}. \quad (33)$$

The coefficient-error vector can now be expressed as

$$\begin{aligned}\Delta\mathbf{w}(k+1) &= \mathbf{P}\{\mathbf{w}(k) - \mu\mathbf{X}(k)[\mathbf{X}^T(k)\mathbf{P}\mathbf{X}(k)]^{-1}\mathbf{X}^T(k)\mathbf{w}(k)\} - \mathbf{w}_{\text{opt}} + \mathbf{F} \\ &= \mathbf{P}\{\Delta\mathbf{w}(k) - \mu\mathbf{X}(k)[\mathbf{X}^T(k)\mathbf{P}\mathbf{X}(k)]^{-1}\mathbf{X}^T(k)\mathbf{w}(k)\} \\ &\quad - (\mathbf{I}-\mathbf{P})\,\mathbf{w}_{\text{opt}} + \mathbf{F} \\ &= \mathbf{P}\{\Delta\mathbf{w}(k) - \mu\mathbf{X}(k)[\mathbf{X}^T(k)\mathbf{P}\mathbf{X}(k)]^{-1}\mathbf{X}^T(k)\mathbf{w}(k)\} \\ &= \{\mathbf{I} - \mu\bar{\mathbf{X}}(k)[\bar{\mathbf{X}}^T(k)\bar{\mathbf{X}}(k)]^{-1}\bar{\mathbf{X}}^T(k)\}\Delta\mathbf{w}(k) - \mu\mathbf{P}\mathbf{X}(k) \\ &\quad \times [\mathbf{X}^T(k)\mathbf{P}\mathbf{X}(k)]^{-1}\mathbf{X}^T(k)\mathbf{w}_{\text{opt}} \\ &= \mathbf{T}(k)\Delta\mathbf{w}(k) - \mu\mathbf{P}\mathbf{X}(k)[\mathbf{X}^T(k)\mathbf{P}\mathbf{X}(k)]^{-1}\mathbf{X}^T(k)\mathbf{w}_{\text{opt}}\end{aligned} \tag{34}$$

where we used $(\mathbf{I}-\mathbf{P})\,\mathbf{w}_{\text{opt}} = \mathbf{F}$. Note that the term $\mu\mathbf{P}\mathbf{X}(k)[\mathbf{X}^T(k)\mathbf{P}\mathbf{X}(k)]^{-1}\mathbf{X}^T(k)\mathbf{w}_{\text{opt}}$ is an additional term as compared with the situation where a training sequence is present. In order to investigate bias, we will study the mean value of the coefficient-error vector given by

$$\begin{aligned}\mathrm{E}[\Delta\mathbf{w}(k+1)] &= \mathrm{E}[\mathbf{T}(k)\Delta\mathbf{w}(k)] \\ &\quad - \mu\mathbf{P}\mathrm{E}\left[\mathbf{X}(k)[\mathbf{X}^T(k)\mathbf{P}\mathbf{X}(k)]^{-1}\mathbf{X}^T(k)\right]\mathbf{w}_{\text{opt}}.\end{aligned} \tag{35}$$

In order to proceed, we use of the following assumption:

A1) *$\Delta\mathbf{w}(k)$ is independent of $\mathbf{P}\mathbf{X}(k)[\mathbf{X}^T(k)\mathbf{P}\mathbf{X}(k)]^{-1}\mathbf{X}^T(k)\mathbf{P}$.*

This assumption is similar to the one made in [31] and [32] for the analysis of the conventional AP algorithm and is a weaker assumption than assuming $\Delta\mathbf{w}(k)$ is independent of $\mathbf{P}\mathbf{X}(k)$ [32]. Introducing the notations

$$\begin{aligned}\bar{\mathbf{T}} &= \mathrm{E}[\mathbf{T}(k)] \\ \bar{\boldsymbol{\xi}} &= -\mu\mathbf{P}\mathrm{E}\left[\mathbf{X}(k)[\mathbf{X}^T(k)\mathbf{P}\mathbf{X}(k)]^{-1}\mathbf{X}^T(k)\right]\mathbf{w}_{\text{opt}}\end{aligned} \tag{36}$$

and using assumption A1) in (35), we have

$$\begin{aligned}\mathrm{E}[\Delta\mathbf{w}(k+1)] &= \bar{\mathbf{T}}\mathrm{E}[\Delta\mathbf{w}(k)] + \bar{\boldsymbol{\xi}} \\ &= \bar{\mathbf{T}}^{k+1}\mathrm{E}[\Delta\mathbf{w}(0)] + \sum_{i=0}^{k}\bar{\mathbf{T}}^i\bar{\boldsymbol{\xi}} \\ &= \left\{\mathbf{I} - \mathrm{E}\left[\mathbf{P}\mathbf{X}(k)[\mathbf{X}^T(k)\mathbf{P}\mathbf{X}(k)]^{-1}\mathbf{X}^T(k)\mathbf{P}\right]\right\}^{k+1} \\ &\quad \times \mathrm{E}[\Delta\mathbf{w}(0)] + \sum_{i=0}^{k}\bar{\mathbf{T}}^i\bar{\boldsymbol{\xi}}.\end{aligned} \tag{37}$$

If we let $k \to \infty$, the first term $\bar{\mathbf{T}}^{k+1}\mathrm{E}[\Delta\mathbf{w}(0)]$ in (37) will go to zero if $0 < \mu < 2$. This is because the eigenvalues of $\bar{\mathbf{T}}$ are bounded as $1-\mu \le \lambda_i \le 1$ (see details in [1]). The second term, which is the bias term, requires some more detailed study. For clarity, let us rewrite the bias term at time instant $k+1$

$$\begin{aligned}\mathbf{b}(k+1) &= \sum_{i=0}^{k}\bar{\mathbf{T}}^i\bar{\boldsymbol{\xi}} \\ &= -\mu\left(\sum_{i=0}^{k}\bar{\mathbf{T}}^i\right)\mathbf{P}\mathrm{E}\left[\mathbf{X}(k)[\mathbf{X}^T(k)\mathbf{P}\mathbf{X}(k)]^{-1}\mathbf{X}^T(k)\right]\mathbf{w}_{\text{opt}}.\end{aligned} \tag{38}$$

For $k \to \infty$, the bias

$$\mathbf{b}(\infty) = -\mu\left(\sum_{i=0}^{\infty}\bar{\mathbf{T}}^i\right)\mathbf{P}\mathrm{E}\left[\mathbf{X}(k)[\mathbf{X}^T(k)\mathbf{P}\mathbf{X}(k)]^{-1}\mathbf{X}^T(k)\right]\mathbf{w}_{\text{opt}}. \tag{39}$$

In order for the bias to go to zero, we need $\mathbf{P}\mathrm{E}\left[\mathbf{X}(k)[\mathbf{X}^T(k)\mathbf{P}\mathbf{X}(k)]^{-1}\mathbf{X}^T(k)\right]\mathbf{w}_{\text{opt}} = \mathbf{0}$ to hold. However, using the expression for $\mathbf{w}_{\text{opt}}$ in (33), this requirement is equivalent to having $\mathrm{E}\left[\mathbf{X}(k)[\mathbf{X}^T(k)\mathbf{P}\mathbf{X}(k)]^{-1}\mathbf{X}^T(k)\right] \propto \mathbf{R}$, which does not hold for the general case. However, when $L = 1$ and N is large, the requirement holds—in this case, the expression $\mathrm{E}\left[\mathbf{X}(k)[\mathbf{X}^T(k)\mathbf{P}\mathbf{X}(k)]^{-1}\mathbf{X}^T(k)\right]$ can be approximated by $\mathbf{R}/\mathrm{tr}\{\mathbf{R}\mathbf{P}\}$—and the bias tends to zero, i.e., when the CAP algorithm is the normalized constrained LMS (NCLMS) algorithm [8], [24]. In other words, we can conclude that normalization may introduce bias for the blind case whenever L is greater than one.

V. Simulation Results

In this section, the performances of the CAP, SM-CAP, and SM-REDCAP algorithms are investigated in three simulations. A first experiment using a desired or reference signal (the more general case), $d(k) \neq 0$, is a system-identification application where the adaptive filter is constrained to have linear phase. The second experiment is a beamforming application where the desired signal is set to zero, i.e., $d(k) = 0$. The last simulation uses the same setup of the first experiment in order to present the performance of the SM-REDCAP.

A. Simulation 1

A simulation was carried out in a system-identification problem where the filter coefficients were constrained to preserve linear phase at every iteration. For this example, we chose $N = 11$ and, in order to fulfill the linear phase requirement, we made

$$\mathbf{C} = \begin{bmatrix}\mathbf{I}_{(N-1)/2} \\ \mathbf{0}^T \\ -\mathbf{J}_{(N-1)/2}\end{bmatrix} \tag{40}$$

with $\mathbf{J}$ being a reversal matrix (an identity matrix with all rows in reversed order), and

$$\mathbf{f} = [0 \cdots 0]^T. \tag{41}$$

This setup was employed to show the improvement of the convergence speed when L is increased. Due to the symmetry of $\mathbf{C}$ and the fact that $\mathbf{f}$ is a null vector, more efficient structures could be used [3]. The input signal consists of colored noise with zero mean, unity variance, and eigenvalue spread around 2068. The reference signal was obtained after filtering the input signal by a linear-phase finite-duration impulse response (FIR) filter and adding observation noise with variance equal to $\sigma_n^2 = 10^{-10}$. The optimal coefficient vector used to compute the coefficient-error vector was obtained from (21) after replacing $\mathbf{R}^{-1}\mathbf{p}$ (the Wiener solution) by $\mathbf{w}_{us}$ (the FIR unknown system). The input signal was taken as colored noise generated by filtering white noise through a filter with a pole at

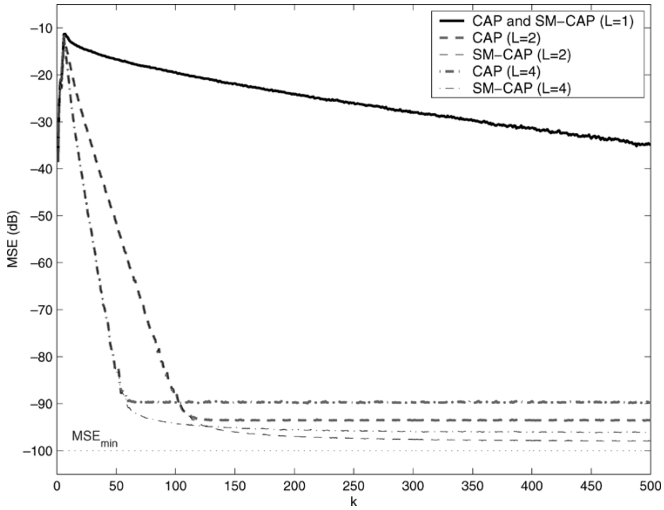

Fig. 2. Learning curves for the CAP and the SM-CAP algorithms with $L = 1$, $L = 2$, and $L = 4$ data reuses, $\sigma_n^2 = 10^{-10}$, $\gamma = \sqrt{6}\sigma_n$, and colored input signal.

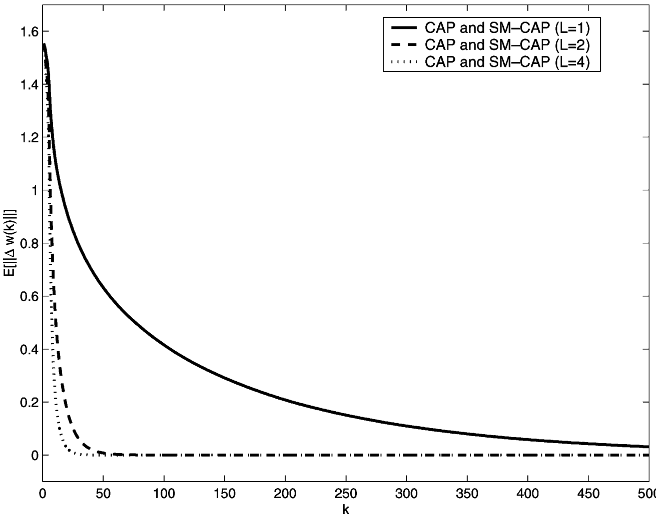

Fig. 3. Evaluating consistency for the CAP and SM-CAP algorithms with $L = 1$, $L = 2$, and $L = 4$ data reuses, $\sigma_n^2 = 10^{-10}$, $\gamma = \sqrt{6}\sigma_n$, and colored input signal (first experiment setup).

$\alpha = 0.99$. The autocorrelation matrix for this example is given by

$$\mathbf{R} = \frac{\sigma_{\mathrm{WGN}}^2}{1-\alpha^2}\begin{bmatrix} 1 & -\alpha & (-\alpha)^2 & \cdots & (-\alpha)^{N-1} \\ -\alpha & 1 & -\alpha & \cdots & (-\alpha)^{N-2} \\ \vdots & \vdots & \ddots & & \vdots \\ (-\alpha)^{N-1} & (-\alpha)^{N-2} & \cdots & & 1 \end{bmatrix} \quad (42)$$

where σ_{WGN}^2 is set such that $\sigma_{\mathrm{WGN}}^2/(1-\alpha^2)$ corresponds to the desired input signal variance σ_x^2, made equal to one in this experiment.

Fig. 2 shows the learning curves for the CAP and the SM-CAP algorithms for $L = 1$, $L = 2$, and $L = 4$. The value of γ in the SM-CAP algorithm was chosen equal to $\sqrt{6}\sigma_n$. A higher value would result in less frequent updates and a possible slower convergence. It is clear from this figure that, for the CAP algorithm, the misadjustment increases with L.[1] It is also clear from this figure that the misadjustment with the SM-CAP algorithm is lower than that of the CAP algorithm and that the misadjustment increases more slowly when L is increased. The only way for the CAP algorithm to achieve the low misadjustment of the SM-CAP algorithm is through the introduction of a step size, resulting in a slower convergence (in this simulation, we have used $\mu = 1$). In cases where the error is very high and fast convergence is required, such as in adaptive beamforming or adaptive multiuser detection in CDMA systems, a variable step size could be used [24] with the CAP algorithm. Furthermore, in 500 iterations, the SM-CAP algorithm performed (in average) updates in 485 (97%), 111 (22.2%), and 100 (20%) time instants for $L = 1$, $L = 2$, and $L = 4$, respectively. In other words, the SM-CAP algorithm with $L = 4$ had a better performance than the CAP algorithm while performing updates for only a fraction of time instants.

[1]Although not visible in Fig. 2, after convergence, the MSE curve corresponding to the CAP algorithm for $L = 1$ is nearly 3 dB lower than the MSE curve corresponding to $L = 2$.

Also for this first experiment, Fig. 3 shows that a consistent estimation of the coefficient vector is obtained, suggesting that there is no bias in the coefficient vector after convergence. In this figure, the CAP algorithm and the SM-CAP algorithm presented basically identical curves.

B. Simulation 2

A second experiment was carried out in a beamforming application. In this scenario, a uniform linear array with $N = 7$ antennas with element spacing equal to a half-wavelength was used in a system with $K = 4$ users. The signal of one user (look-direction set to 0°) is of interest, and the other three signals (incident angles corresponding to $-25°$, 45 °, and 50°) are treated as interferers or *jammers*. The received discrete-time signal can be modeled as

$$\mathbf{x}(k) = \mathbf{SAu}(k) + \mathbf{n}(k)$$

where $\mathbf{S} = [\mathbf{s}(\theta_1)\ \mathbf{s}(\theta_2)\ \ldots\ \mathbf{s}(\theta_K)]$ is the *steering matrix* containing the steering vectors of the users, θ_i is the DOA, $\mathbf{A} = \mathrm{diag}[A_1\ A_2\ \ldots\ A_K]$ contains the user amplitudes A_i, $\mathbf{u}(k) = [u_1(k)\,u_2(k)\ \ldots\ u_K(k)]^T$ is a vector of the transmitted user information, and $\mathbf{n}(k)$ is the sampled noise sequence across the array. The *signal-to-noise* ratio (SNR) was set to 0 dB, and *jammer-to-noise* ratios (JNRs) of 30 dB were used.

The learning curves—actually the mean output energy (MOE) for this case of no training signal—are depicted in Fig. 4. We observe, from this figure, approximately the same behavior as in the first experiment: a lower misadjustment for the SM-CAP algorithm.

In Fig. 5, the norm of the average coefficient-error norm is shown, suggesting that there is bias for large values of L.

In this beamforming simulation, we have also plotted the beam pattern for the CAP and the SM-CAP algorithms. It was done in two time instants and using the results of one single

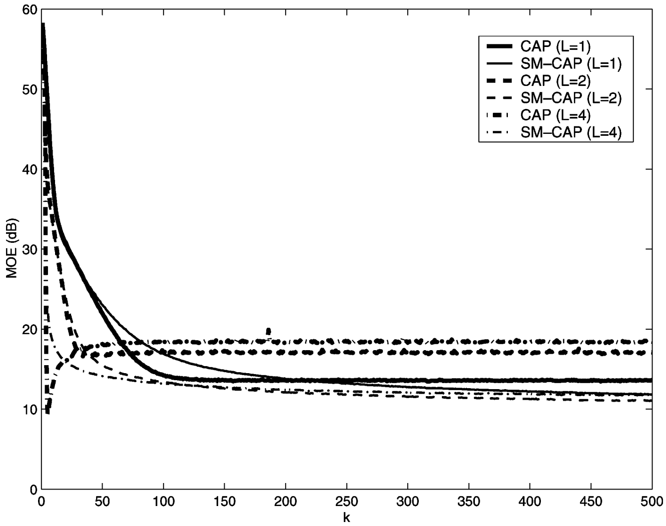

Fig. 4. Learning curves for the CAP and the SM-CAP algorithms with $L = 1$, $L = 2$, and $L = 4$ data reuses, $\gamma = \sqrt{48}$, SNR = 0 dB, and JNR = 30 dB in a beamforming application.

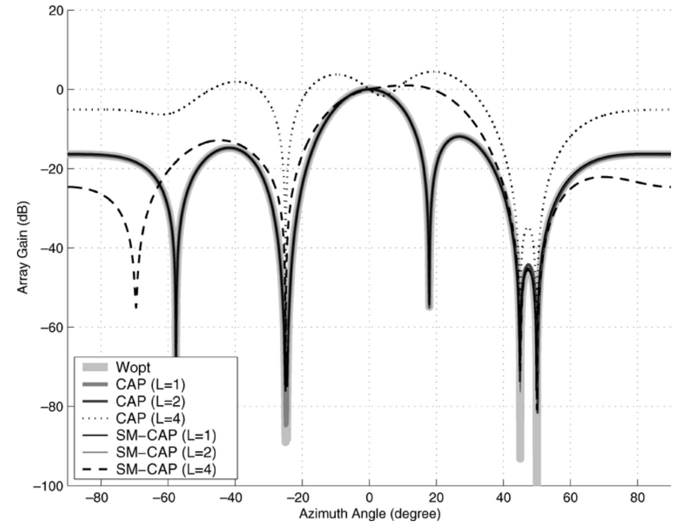

Fig. 6. Beamforming patterns for the CAP and the SM-CAP algorithms with $L = 1$, $L = 2$, and $L = 4$ data reuses, SNR = 0 dB, and JNR = 30 dB in a beamforming application after 500 iterations.

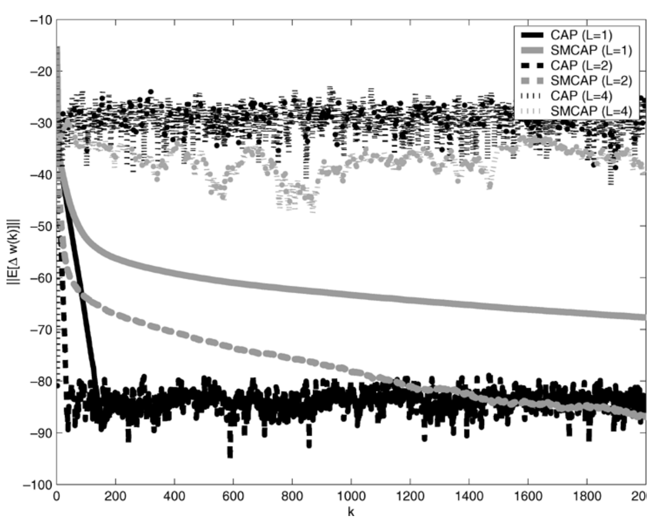

Fig. 5. Average coefficient-error norm for the CAP and the SM-CAP algorithms with $L = 1$, $L = 2$, and $L = 4$ data reuses, $\gamma = \sqrt{48}$, SNR = 0 dB, and JNR = 30 dB in a beamforming application.

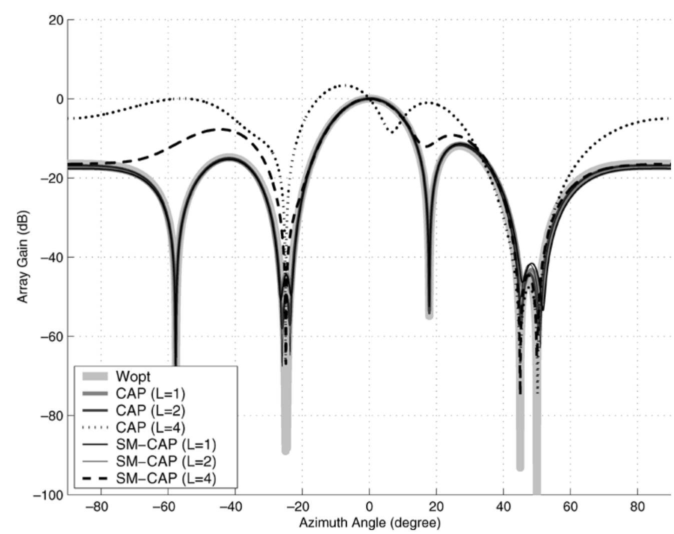

Fig. 7. Beamforming patterns for the CAP and the SM-CAP algorithms with $L = 1$, $L = 2$, and $L = 4$ data reuses, SNR = 0 dB, and JNR = 30 dB in a beamforming application after 20 iterations.

run. Fig. 6 shows the results after a large (500) number of samples, and Fig. 7 shows the results after a small (20) number of samples. From both figures, we observe that algorithms CAP and SM-CAP, due to bias in blind applications, do not present a good performance in terms of beam pattern for large number of data reuses L. Conversely, the implementations with a small number of data reuses, $L = 1$ or 2, render very good sample support capability.

C. Simulation 3

In this simulation, we have used the same setup of the first experiment to test the performance of the SM-REDCAP algorithm.

The result of this simulation in terms of learning curves are depicted in Fig. 8. We see from this figure that the SM-REDCAP algorithm performs very similarly to the SM-CAP algorithm for all values of data reuses (L) employed. It is only noticeable a slightly lower misadjustment (lower MSE after convergence) in the SM-CAP learning curves.

VI. Conclusion

In this paper, we have introduced the constrained affine-projection (CAP) algorithm as well as two set membership CAP algorithms. These data-selective versions of the CAP algorithm can, in certain applications, substantially reduce the number of required updates.

Through theoretical analysis, we have shown that these algorithms may present bias in the coefficient vector. Simulation results of two experiments, including both cases of biased and unbiased solutions, supported the analysis claims and evaluated

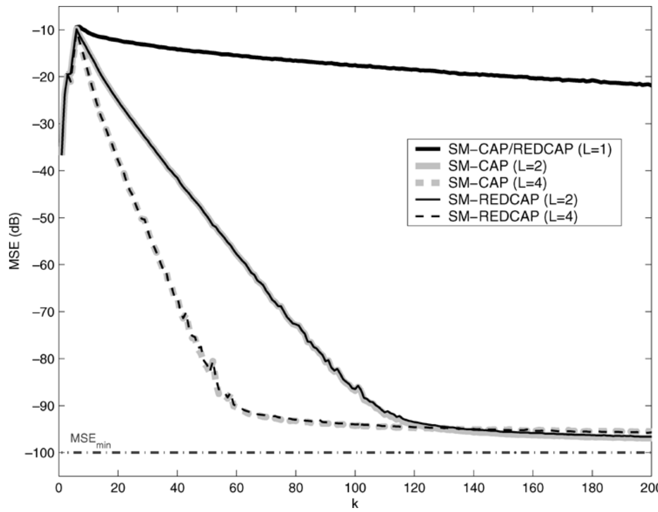

Fig. 8. Learning curves for the SM-CAP and the SM-REDCAP algorithms with $L = 1$, $L = 2$, and $L = 4$ data reuses, $\sigma_n^2 = 10^{-10}$, $\gamma = \sqrt{6}\sigma_n$, and colored input signal.

the performance of the proposed algorithms. A third simulation shows the very good performance achieved with a much simplified version of the set-membership CAP (SM-CAP) algorithm, namely the set-membership reduced peak-complexity CAP (SM-REDCAP) algorithm, first developed and presented here.

REFERENCES

[1] O. L. Frost III, "An algorithm for linearly constrained adaptive array processing," *Proc. IEEE*, vol. 60, no. 8, pp. 926–935, Aug. 1972.

[2] M. Honig, U. Madhow, and S. Verdú, "Blind multiuser detection," *IEEE Trans. Inf. Theory*, vol. 41, no. 4, pp. 944–960, Jul. 1995.

[3] L. S. Resende, J. M. T. Romano, and M. G. Bellanger, "Simplified FLS algorithm for linear phase adaptive filtering," in *Proc. Eur. Signal Processing Conf.*, vol. 3, Rhodes, Greece, 1998, pp. 1237–1240.

[4] L. J. Griffiths and C. W. Jim, "An alternative approach to linearly constrained adaptive beamforming," *IEEE Trans. Antennas and Propag.*, vol. AP-30, no. 1, pp. 27–34, Jan. 1982.

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

[6] Z. Tian, K. L. Bell, and H. L. Van Trees, "Robust constrained linear receivers for CDMA wireless systems," *IEEE Trans. Signal Process.*, vol. 49, no. 7, pp. 1510–1522, Jul. 2001.

[7] M. L. R. de Campos, S. Werner, and J. A. Apolinário Jr., "Householder-transform constrained LMS algorithms with reduced-rank updating," in *Proc. IEEE Int. Conf. Acoustics, Speech, Signal Processing*, vol. 4, Phoenix, AZ, Mar. 1999, pp. 1857–1860.

[8] ——, "Constrained adaptation algorithms employing householder transformation," *IEEE Trans. Signal Process.*, vol. 50, no. 9, pp. 2187–2195, Sep. 2002.

[9] L. S. Resende, J. M. T. Romano, and M. G. Bellanger, "A fast least-squares algorithm for linearly constrained adaptive filtering," *IEEE Trans. Signal Process.*, vol. 44, no. 5, pp. 1168–1174, May 1996.

[10] K. Ozeki and T. Umeda, "An adaptive filtering algorithm using an orthogonal projection to an affine subspace and its properties," *Electron. Commun. Jpn.*, vol. 67-A, pp. 126–132, 1984.

[11] S. G. Sankaran and A. A. Beex, "Convergence behavior of affine projection algorithms," *IEEE Trans. Signal Process.*, vol. 48, no. 4, pp. 1086–1096, Apr. 2000.

[12] M. Rupp, "A family of adaptive filtering algorithms with decorrelating properties," *IEEE Trans. Signal Process.*, vol. 46, no. 3, pp. 771–775, Mar. 1998.

[13] M. L. R. de Campos, J. A. Apolinário Jr., and P. S. R. Diniz, "On normalized data-reusing and affine projection algorithms," in *Proc. IEEE Int. Conf. Electronics, Circuits, Systems (ICECS'99)*, Pafos, Cyprus, Sep. 1999, pp. 843–846.

[14] M. Montazeri and P. Duhamel, "A set of algorithms linking NLMS and block RLS algorithms," *IEEE Trans. Signal Process.*, vol. 43, no. 2, pp. 444–453, Feb. 1995.

[15] S. Gollamudi, S. Nagaraj, and Y. F. Huang, "SMART: A toolbox for set-membership filtering," in *Proc. Eur. Conf. Circuit Theory Design*, Aug. 1997, pp. 879–884.

[16] S. Gollamudi, S. Nagaraj, S. Kapoor, and Y. F. Huang, "Set-membership filtering and a set-membership normalized LMS algorithm with an adaptive step size," *IEEE Signal Process. Lett.*, vol. 5, no. 5, pp. 111–114, May 1998.

[17] ——, "Set-membership adaptive equalization and updator-shared implementation for multiple channel communications systems," *IEEE Trans. Signal Process.*, vol. 46, no. 9, pp. 2372–2384, Sep. 1998.

[18] P. S. R. Diniz and S. Werner, "Set-membership binormalized data-reusing algorithms," in *Proc. IFAC Symp. System Identification (SYSID 2000)*, vol. 3, Santa Barbara, CA, Jun. 2000, pp. 869–874.

[19] ——, "Set-membership binormalized data-reusing algorithms," *IEEE Trans. Signal Process.*, vol. 51, no. 1, pp. 124–134, Jan. 2003.

[20] S. Werner and P. S. R. Diniz, "Set-membership affine projection algorithm," *Signal Process. Lett.*, vol. 8, no. 8, pp. 231–235, Aug. 2001.

[21] S. Werner, "Reduced complexity adaptive filtering algorithms with applications to communications systems," D.Sc. dissertation, Department of Electrical Engineering, Helsinki University of Technology Signal Processing Laboratory, Espoo, Finland, 2002.

[22] S. Nagaraj, S. Gollamudi, S. Kapoor, Y. F. Huang, and J. R. Deller, "Adaptive interference suppression for CDMA systems with a worst-case error criterion," *IEEE Trans. Signal Process.*, vol. 48, no. 1, pp. 284–289, Jan. 2000.

[23] K. H. Afkhamie, Z.-Q. Luo, and K. M. Wong, "Adaptive linear filtering using interior point optimization techniques," *IEEE Trans. Signal Process.*, vol. 48, no. 6, pp. 1637–1648, Jun. 2000.

[24] J. A. Apolinário Jr., S. Werner, and P. S. R. Diniz, "Constrained normalized adaptive filters for CDMA mobile communications," in *Proc. Eur. Signal Processing Conf.*, vol. 4, Rhodes, Greece, 1998, pp. 2053–2056.

[25] M. L. R. de Campos and J. A. Apolinário Jr., "The constrained affine projection algorithm – Development and convergence issues," presented at the 1st Balkan Conf. Signal Processing, Communications, Circuits, Systems, Istanbul, Turkey, May 2000.

[26] S. Werner, M. L. R. de Campos, and J. A. Apolinário Jr., "Data-selective constrained affine projection algorithm," in *Proc. IEEE Int. Conf. Acoustics, Speech, Signal Processing (ICASSP'01)*, vol. 6, Salt Lake City, UT, May 2001, pp. 3745–3748.

[27] J. R. Deller, S. Gollamudi, S. Nagaraj, and Y. F. Huang, "Convergence analysis of the QUASI-OBE algorithm and the performance implications," in *Proc. IFAC Int. Symp. System Identification (SYSID 2000)*, vol. 3, Santa Barbara, CA, Jun. 2000, pp. 875–880.

[28] S. Nagaraj, S. Gollamudi, S. Kapoor, and Y. F. Huang, "BEACON: An adaptive set-membership filtering technique with sparse updates," *IEEE Trans. Signal Process.*, vol. 47, no. 11, pp. 2928–2941, Nov. 1999.

[29] M. T. Schiavoni and M. G. Amin, "A linearly constrained minimization approach to adaptive linear phase and notch filters," in *Proc. 20th Southeastern Symp. System Theory*, Philadelphia, PA, Mar. 1988, pp. 682–685.

[30] M. L. R. de Campos, S. Werner, and J. A. Apolinário Jr., "Constrained adaptive filters," in *Adaptive Antenna Arrays: Trends and Applications*, S. Chrandran, Ed. New York: Springer-Verlag, 2004.

[31] H.-C. Shin and A. H. Sayed, "Transient behavior affine projection algorithms," in *Proc. IEEE Int. Conf. Acoustics, Speech, Signal Processing (ICASSP'03)*, vol. 6, Hong Kong, Apr. 2003, pp. 353–356.

[32] ——, "Mean-square performance of affine projection algorithms," *IEEE Trans. Signal Process.*, vol. 52, no. 1, pp. 90–102, Jan. 2004.

Stefan Werner (M'03) received the M.Sc. degree in electrical engineering from the Royal Institute of Technology, Stockholm, Sweden, in 1998 and the D.Sc. (EE) degree (with honors) from the Signal Processing Laboratory, Smart and Novel Radios (SMARAD) Center of Excellence, Helsinki University of Technology (HUT), Espoo, Finland, in 2002.

He is currently a Senior Researcher with the Signal Processing Laboratory, HUT. His research interests are in multiuser communications and adaptive filtering.

José Antonio Apolinário Jr. (SM'04) was born in Taubaté, Brazil, in January 1960. He graduated from the Military Academy of Agulhas Negras (AMAN), Resende, Brazil, in 1981 and received the B.Sc. degree from the Military Institute of Engineering (IME), Rio de Janeiro, Brazil, in 1988, the M. Sc. degree from the University of Brasília (UnB), Brasília, Brazil, in 1993, and the D.Sc. degree from the Federal University of Rio de Janeiro (COPPE/UFRJ), Rio de Janeiro, Brazil, in 1998, all in electrical engineering.

He is currently with the IME, heading the Department of Electrical Engineering. He was a Visiting Professor at the Escuela Politécnica del Ejército, Quito, Ecuador, in 1999 and 2000 and a Visiting Researcher and Visiting Professor at the Helsinki University of Technology, Finland, in 1997 and 2004, respectively. His research interests comprise many aspects of linear and nonlinear digital signal processing, including adaptive systems and speech and array processing.

Dr. Apolinário has organized the Rio de Janeiro Chapter of the IEEE Communications Society, where he served as the first Chapter Chair.

Marcello L. R. de Campos (SM'02) was born in Niteroi, Brazil, in 1968. He received the B.Sc. degree (*cum laude*) from the Federal University of Rio de Janeiro (UFRJ), Rio de Janeiro, Brazil, in 1990, the M.Sc. degree from COPPE/UFRJ in 1991, and the Ph.D. degree from the University of Victoria, Victoria, BC, Canada, in 1995, all in electrical engineering.

In 1996, he was Postdoctoral Fellow with the Department of Electronics, School of Engineering, UFRJ, and with the Program of Electrical Engineering, COPPE/UFRJ. From January 1997 until May 1998, he was Associate Professor with the Department of Electrical Engineering (DE/3), Military Institute of Engineering (IME), Rio de Janeiro. He is currently Associate Professor and Chair of the Program of Electrical Engineering, COPPE/UFRJ. From September to December 1998, he was visiting the Laboratory for Telecommunications Technology, Helsinki University of Technology, Espoo, Finland. In 2001, he received a Nokia Visiting Fellowship to visit the Centre for Wireless Communications, University of Oulu, Oulu, Finland. His research interests include adaptive signal processing, statistical signal processing, and signal processing for communications.

Dr. de Campos served as IEEE Communications Society Regional Director for Latin America in 2000 and 2001.

Paulo S. R. Diniz (F'00) was born in Niterói, Brazil. He received the Electronics Engineering degree (*cum laude*) from the Federal University of Rio de Janeiro (UFRJ), Rio de Janeiro, Brazil, in 1978, the M.Sc. degree from COPPE/UFRJ in 1981, and the Ph.D. from Concordia University, Montreal, QC, Canada, in 1984, all in electrical engineering.

Since 1979, he has been with the Department of Electronic Engineering (the undergraduate department) of UFRJ. He has also been with the Program of Electrical Engineering (the graduate studies department), COPPE/UFRJ, since 1984, where he is presently a Professor. He served as Undergraduate Course Coordinator and as Chairman of the Graduate Department. He is one of the three senior researchers and coordinators of the National Excellence Center in Signal Processing. From January 1991 to July 1992, he was a visiting Research Associate in the Department of Electrical and Computer Engineering of University of Victoria, Victoria, B.C., Canada. He also holds a Docent position at Helsinki University of Technology. From January 2002 to June 2002, he was a Melchor Chair Professor in the Department of Electrical Engineering of University of Notre Dame, Notre Dame, IN. His teaching and research interests are in analog and digital signal processing, adaptive signal processing, digital communications, wireless communications, multirate systems, stochastic processes, and electronic circuits. He has published several refereed papers in some of these areas and wrote the book *Adaptive Filtering: Algorithms and Practical Implementation* (Norwell, MA: Kluwer, 2nd ed., 2002) and coauthored the book *Digital Signal Processing: System Analysis and Design* (Cambridge, U.K.: Cambridge Univ. Press, 2002) with E. A. B. da Silva and S. L. Netto.

Dr. Diniz was the Technical Program Chair of the 1995 Midwest Symposium on Circuits and Systems (MWSCAS) held in Rio de Janeiro, Brazil. He has been on the technical committee of several international conferences. He has served as Vice President for Region 9 of the IEEE Circuits and Systems Society and as Chairman of the Digital Signal Processing technical committee of the same Society. He has served as Associate Editor for the following journals: IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS II: ANALOG AND DIGITAL SIGNAL PROCESSING from 1996 to 1999, IEEE TRANSACTIONS ON SIGNAL PROCESSING from 1999 to 2002, and the *Circuits, Systems and Signal Processing Journal* from 1998 to 2002. He was a Distinguished Lecturer of the IEEE Circuits and Systems Society for the year 2000 to 2001. In 2004, he served as Distinguished Lecturer of the IEEE Signal Processing Society and received the 2004 Education Award of the IEEE Circuits and Systems Society. He also has received the Rio de Janeiro State Scientist Award from the Governor of Rio de Janeiro state.