Constrained Adaptation Algorithms Employing Householder Transformation

Marcello L. R. de Campos, *Member, IEEE*, Stefan Werner, *Student Member, IEEE*, and José Antonio Apolinário, Jr., *Member, IEEE*

***Abstract*—This paper presents a tutorial-like detailed explanation of linearly constrained minimum-variance filtering in order to introduce an efficient implementation that utilizes Householder transformation (HT). Through a graphical description of the algorithms, further insight on linearly constrained adaptive filters was made possible, and the main differences among several algorithms were highlighted. The method proposed herein, based on HT, allows direct application of any unconstrained adaptation algorithm as in a generalized sidelobe canceller (GSC), but unlike the GSC, the HT-based approach always renders efficient implementations. A complete and detailed comparison with the GSC model and a thorough discussion of the advantages of the HT-based approach are also given. Simulations were run in a beamforming application where a linear array of 12 sensors was used. It was verified that not only the HT approach yields efficient implementation of constrained adaptive filters, but in addition, the beampatterns achieved with this method were much closer to the optimal solution than the beampatterns obtained with GSC models with similar computational complexity.**

***Index Terms*—Adaptation algorithms, beamforming, efficient alogorithms, generalized sidelobe canceller, linearly constrained adaptive filters.**

I. Introduction

ADAPTIVE receiving-antenna systems that can operate in real-time were developed in the 1960s [1], [2] and were intended to perform directional and spatial filtering with minimum knowledge of the statistics of arriving signals. Linearly constrained (LC) adaptive array processing [3] was undoubtedly a significant improvement to previously devised adaptive antenna-array systems for the need of training sequences and knowledge of interfering-signal statistics became unnecessary. In the approach presented in [3], output power is minimized, whereas a desired signal arriving at a known direction is linearly filtered according to a specified frequency response.

Manuscript received May 7, 2001; revised May 5, 2002. The associate editor coordinating the review of this paper and approving it for publication was Dr. Naofal M. W. Al-Dhahir.

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

S. Werner is with the Signal Processing Laboratory, Helsinki University of Technology, Espoo, 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, Brazil (e-mail: apolin@ieee.org).

Publisher Item Identifier 10.1109/TSP.2002.801893.

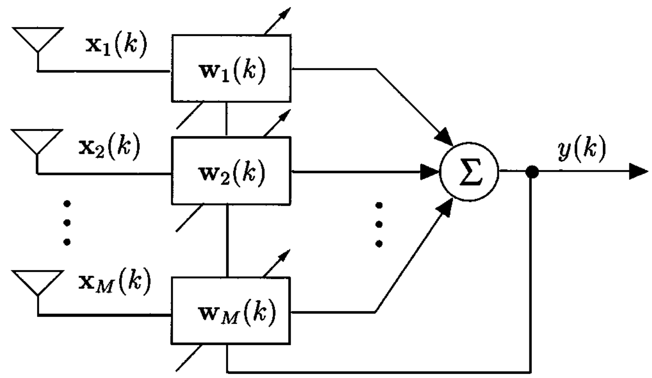

Fig. 1. Broadband adaptive receiving array.

Fig. 1 shows a schematic diagram of a broadband array-processing filter with M sensors and filters with N taps. The output of the array may be expressed as $y(k) = \mathbf{w}^{\mathrm{T}}(k)\mathbf{x}(k)$, where

$$\mathbf{w}(k) = \left[\mathbf{w}_1^{\mathrm{T}}(k)\ \ \mathbf{w}_2^{\mathrm{T}}(k)\ \ \cdots\ \ \mathbf{w}_M^{\mathrm{T}}(k)\right]^{\mathrm{T}} \tag{1}$$

$$\mathbf{x}(k) = \left[\mathbf{x}_1^{\mathrm{T}}(k)\ \ \mathbf{x}_2^{\mathrm{T}}(k)\ \ \cdots\ \ \mathbf{x}_M^{\mathrm{T}}(k)\right]^{\mathrm{T}} \tag{2}$$

and

$$\mathbf{x}_i(k) = \left[x_i(k)\ \ x_i(k-1)\ \ \cdots\ \ x_i(k-N+1)\right]^{\mathrm{T}}. \tag{3}$$

For LC adaptive filters, the coefficient update is performed in a subspace that is orthogonal to the subspace spanned by a constraint matrix [3]. The direction of the update is given by the input-signal vector premultiplied by a projection matrix, which is rank-deficient (cf. Section II). This may be regarded as the use of nonpersistently exciting input signal,[1] and lack of persistence of excitation requires that a correction term be added to the coefficients to prevent accumulation of roundoff errors [3], [5].

An alternative approach to the implementation of LC array processing was introduced by Griffiths and Jim in [5], which became known as the generalized sidelobe canceling (GSC) model. With the GSC model, the dimension of the adaptation subspace is properly reduced by means of a blocking matrix such that persistence of excitation is not lost due to imposing constraints. By transforming the constrained minimization problem into an unconstrained minimization problem, the GSC model allows that any adaptation algorithm can be directly applied. Furthermore, as the restriction imposed on the blocking matrix is only that its columns must be orthogonal to the constraint matrix, a myriad of possible implementations result.

[1]For a discussion on persistence of excitation, see [4].

The structure of the blocking matrix has a direct effect on the overall computational complexity through its multiplication to the input-signal vector in each iteration. Construction of the blocking matrix using, e.g., singular-value decomposition (SVD), results in a matrix with no special structure, which in the GSC model renders high computational complexity per iteration. In these cases, the practical use of the GSC structure is questionable. The extra computations resulting from the product of the blocking matrix by the input-signal vector may exceed those of the adaptation algorithm and filtering operation by up to one order of magnitude. Other types of blocking matrices with sparse structures may be of more practical use from the perspective of computational complexity. Many times, such solutions are application dependent, and the resulting matrix is, in general, not orthogonal; in these cases, adaptive implementations of the GSC and linearly constrained minimum-variance (LCMV) filters may bear no relation [5].

The main contributions of this paper are concerned with an efficient implementation of LC adaptive filtering algorithms that overcomes the problem of added computational complexity that may occur in the GSC structure. In addition, as in the GSC structure, the proposed method may be used with any unconstrained adaptive filtering algorithm. A geometrical interpretation is presented in order to provide better understanding of the updating process used in constrained algorithms. Moreover, this geometrical interpretation is used to illustrate better the use of the Householder transformation. A detailed explanation of the matrices involved in the process is presented, and pseudo-code routines are provided. Even in cases where the GSC structure is equivalent to the Householder implementation introduced here, the latter is more efficient.

The organization of the paper is as follows. Section II gives the background of the LCMV filter and the GSC model. In Section III, the new Householder-transform constrained algorithms are presented as a low-complexity solution for reducing the subspace in which adaptive-filter coefficients are updated. Relations to the GSC model are made, resulting in a framework where any unconstrained adaptation algorithm can be applied to linearly constrained problems using the proposed method. Section IV contains simulation results, followed by conclusions in Section V.

II. Background

A. Optimal Linearly-Constrained Minimum-Variance Filter

The optimal (LCMV) filter in the sense of the minimum mean output energy (MOE) is the one that minimizes the objective function

$$\xi_w = \frac{1}{2}\left\|\mathbf{R}^{1/2}\mathbf{w}\right\|^2 = \frac{1}{2}\,\mathbf{w}^{\mathrm{T}}\mathbf{R}\mathbf{w} \tag{4}$$

subjected to the set of linear constraints defined by

$$\mathbf{C}^{\mathrm{T}}\mathbf{w} = \mathbf{f} \tag{5}$$

where

- $\mathbf{w}$ vector of coefficients of length MN;
- $\mathbf{R}^{1/2}$ square-root factor of the autocorrelation matrix of the input signal;
- $\mathbf{C}$ $MN \times p$ constraint matrix;
- $\mathbf{f}$ $p \times 1$ gain vector.

Note that p corresponds to the number of constraints.

By using the method of Lagrange multipliers, the optimal solution becomes [3], [6]

$$\mathbf{w}_{opt} = \mathbf{R}^{-1}\mathbf{C}\left(\mathbf{C}^{\mathrm{T}}\mathbf{R}^{-1}\mathbf{C}\right)^{-1}\mathbf{f}. \tag{6}$$

The equation above bears the difficulty of knowing in real time the inverse of the input-signal autocorrelation matrix $\mathbf{R}^{-1}$. A much more practical approach is to produce an estimate of $\mathbf{w}_{opt}$ recursively at every iteration. As time proceeds, the estimate is improved such that convergence in the mean to the optimal solution may eventually be achieved.

B. LC Adaptive Filtering

Frost [3] has proposed an algorithm to estimate $\mathbf{w}_{opt}$ based on the gradient method or, more specifically, based on the least-mean-square (LMS) algorithm for adaptive filtering. Let $\mathbf{w}(k)$ denote the estimate of $\mathbf{w}_{opt}$ at time instant k and

$$y(k) = \mathbf{w}^{\mathrm{T}}(k)\mathbf{x}(k) \tag{7}$$

denote the filter output, equal in absolute value to the output error, since in this case, the reference signal is zero.

The constrained LMS (CLMS) algorithm [3] uses as an estimate of the input-signal autocorrelation matrix $\mathbf{R}$ at instant k the outer product of the input-signal vector by itself, i.e., $\hat{\mathbf{R}} = \mathbf{x}(k)\mathbf{x}^{\mathrm{T}}(k)$. In this case, the coefficient-update equation becomes [3]

$$\begin{aligned}\mathbf{w}(k+1) &= \mathbf{w}(k) - \mu y(k)\left[\mathbf{I} - \mathbf{C}\left(\mathbf{C}^{\mathrm{T}}\mathbf{C}\right)^{-1}\mathbf{C}^{\mathrm{T}}\right]\mathbf{x}(k)\\ &\quad + \mathbf{C}\left(\mathbf{C}^{\mathrm{T}}\mathbf{C}\right)^{-1}\left[\mathbf{f} - \mathbf{C}^{\mathrm{T}}\mathbf{w}(k)\right]\\ &= \mathbf{w}(k) - \mu y(k)\mathbf{P}\mathbf{x}(k)\\ &\quad + \mathbf{C}\left(\mathbf{C}^{\mathrm{T}}\mathbf{C}\right)^{-1}\left[\mathbf{f} - \mathbf{C}^{\mathrm{T}}\mathbf{w}(k)\right]\\ &= \mathbf{P}[\mathbf{w}(k) - \mu y(k)\mathbf{x}(k)] + \mathbf{F}\end{aligned} \tag{8}$$

where $\mathbf{I}$ is the MNth-order identity matrix

$$\mathbf{P} = \mathbf{I} - \mathbf{C}\left(\mathbf{C}^{\mathrm{T}}\mathbf{C}\right)^{-1}\mathbf{C}^{\mathrm{T}} \tag{9}$$

is the projection matrix onto the subspace orthogonal to the subspace spanned by the constraint matrix, and [3]

$$\mathbf{F} = \mathbf{C}\left(\mathbf{C}^{\mathrm{T}}\mathbf{C}\right)^{-1}\mathbf{f}. \tag{10}$$

Note in (8) that the term multiplied by the projection matrix $\mathbf{w}(k) - \mu y(k)\mathbf{x}(k)$ corresponds to the unconstrained LMS solution, which is projected onto the homogeneous hyperplane $\mathbf{C}^{\mathrm{T}}\mathbf{w} = \mathbf{0}$ and moved back to the constraint hyperplane by adding vector $\mathbf{F}$. Fig. 2 illustrates this operation.

A normalized version of the CLMS algorithm, namely, the NCLMS algorithm, can be easily derived [7]; the update equation becomes

$$\begin{aligned}\mathbf{w}(k+1) &= \mathbf{P}\left[\mathbf{w}(k) - \mu\frac{y(k)}{\mathbf{x}^{\mathrm{T}}(k)\mathbf{P}\mathbf{x}(k)}\,\mathbf{x}(k)\right] + \mathbf{F}\\ &= \mathbf{w}(k) - \mu\frac{y(k)}{\mathbf{x}^{\mathrm{T}}(k)\mathbf{P}\mathbf{x}(k)}\,\mathbf{P}\mathbf{x}(k)\\ &\quad + \mathbf{C}\left(\mathbf{C}^{\mathrm{T}}\mathbf{C}\right)^{-1}\left[\mathbf{f} - \mathbf{C}^{\mathrm{T}}\mathbf{w}(k)\right].\end{aligned} \tag{11}$$

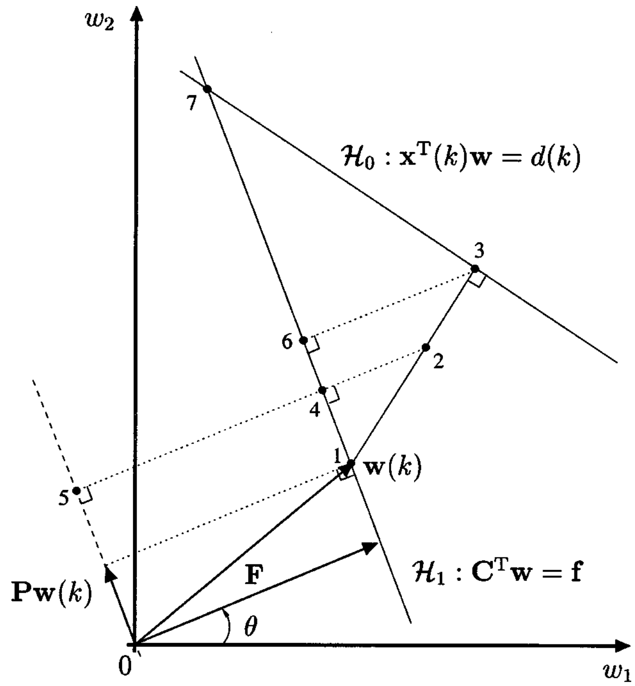

Fig. 2. Geometrical interpretation of some constrained algorithms. 1) $\mathbf{w}(k) = \mathbf{F} + \mathbf{Pw}(k)$. 2) $\mathbf{w}(k+1)$ for the unconstrained LMS algorithm. 3) $\mathbf{w}(k+1)$ for the unconstrained NLMS algorithm. 4) $\mathbf{w}(k+1)$ for the constrained LMS algorithm [3]. 5) $\mathbf{Pw}(k+1)$ for the constrained LMS algorithm [3]. 6) $\mathbf{w}(k+1)$ for the projected NLMS algorithm [8]. 7) $\mathbf{w}(k+1)$ for the constrained NLMS algorithm [7].

We will stress here the fact that for the normalized constrained LMS (NCLMS) algorithm in (11), the *a posteriori* output signal is zero for $\mu = 1$. The solution is at the intersection of the hyperplane defined by the constraints $\mathcal{H}_1$: $\mathbf{C}^{\mathrm{T}}\mathbf{w} = \mathbf{f}$ with the hyperplane defined by the null *a posteriori* condition $\mathcal{H}_0$: $\mathbf{x}^{\mathrm{T}}(k)\mathbf{w} = d(k) = 0$. Therefore, the solution $\mathbf{w}(k+1)$ is not merely a projection of the solution of the normalized LMS (NLMS) algorithm onto the hyperplane defined by the constraints. This is also illustrated in Fig. 2, where we present a detailed graphical description of the coefficient update of several algorithms in the case of two coefficients only. In this case, hyperplanes $\mathcal{H}_0$ and $\mathcal{H}_1$ become two lines, as noted in the figure. As $\mathbf{w}(k)$ must satisfy the constraints, it must belong to $\mathcal{H}_1$ and can be decomposed in two mutually orthogonal vectors $\mathbf{F}$ and $\mathbf{Pw}(k)$. The figure also illustrates how the solutions of the constrained version of the LMS algorithm, the NLMS algorithm, and the projection algorithm [8] relate. Note that in this figure, all updated vectors for the constrained algorithms are located along the direction of $\mathbf{C}^{\mathrm{T}}\mathbf{w} = \mathbf{f}$ (points 4, 6, and 7). Therefore, if $\mathbf{w}$ were rotated such that the w_1 axis and $\mathbf{F}$ had the same direction, the component along this direction would not need to be updated. This fact will be used in Section III, where the new Householder-transform algorithms are introduced.

The necessity of the last term in (8) and (11) may be surprising for it is expected that all $\mathbf{w}(k)$ satisfy the constraint, and therefore, this last term should be equal to zero. In practical implementations, however, this term will be included to prevent divergence in a limited-precision arithmetic machine [5] due to perturbations introduced in the coefficient vector in a direction that is not excited by vector $\mathbf{Px}(k)$. The same reasoning can be applied to the constrained recursive least-squares (CRLS) algorithm presented in [9] and to the constrained quasi-Newton (CQN) algorithm presented in [10].

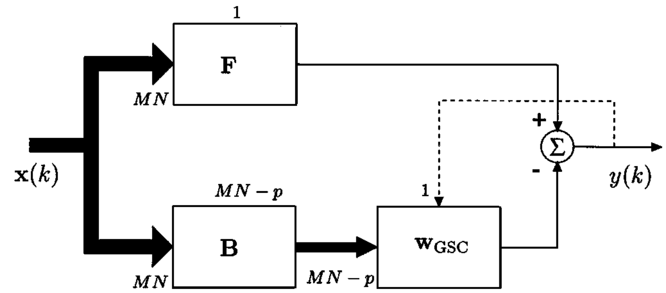

Fig. 3. Generalized sidelobe canceling (GSC) model.

C. Generalized Sidelobe Canceling Model

Many implementations of LC adaptive filters utilize the advantages of the GSC model [6], mainly, because this model employs unconstrained adaptation algorithms that have been extensively studied in the literature. Fig. 3 shows the schematic of the GSC model.

Let $\mathbf{B}$ in Fig. 3 be a full-rank $MN \times (MN-p)$ blocking matrix designed to completely filter out the components of the input signal that are in the same direction as the constraints. Matrix $\mathbf{B}$ must span the null space of the constraint matrix $\mathbf{C}$, i.e.,

$$\mathbf{B}^{\mathrm{T}}\mathbf{C} = \mathbf{0}. \tag{12}$$

In order to relate the GSC model and the linearly constrained minimum variance (LCMV) filter, let $\mathbf{T}$ be an $MN \times MN$ transformation matrix such that

$$\mathbf{T} = [\mathbf{C} \quad \mathbf{B}]. \tag{13}$$

Now, suppose that a transformed coefficient vector $\overline{\mathbf{w}}$ relates to the LCMV coefficient vector $\mathbf{w}$ through

$$\mathbf{w} = \mathbf{T}\overline{\mathbf{w}}. \tag{14}$$

This transformation of the coefficient vector does not modify the output error [11] as long as $\mathbf{T}$ is invertible, which is always guaranteed from (12). If we partition vector $\overline{\mathbf{w}}$ as

$$\overline{\mathbf{w}} = \begin{bmatrix} \overline{\mathbf{w}}_U \\ -\overline{\mathbf{w}}_L \end{bmatrix} \tag{15}$$

with $\overline{\mathbf{w}}_U$ and $\overline{\mathbf{w}}_L$ vectors of dimensions $p \times 1$ and $(MN-p) \times 1$, respectively, it can be easily shown that we have [6]

$$\mathbf{w} = \mathbf{F} - \mathbf{B}\overline{\mathbf{w}}_L \tag{16}$$

and

$$\overline{\mathbf{w}}_U = (\mathbf{C}^{\mathrm{T}}\mathbf{C})^{-1}\mathbf{f} \tag{17}$$

$$\overline{\mathbf{w}}_L = \mathbf{w}_{GSC} \tag{18}$$

where $\mathbf{F} = \mathbf{C}\overline{\mathbf{w}}_U$ is the constant part of vector $\overline{\mathbf{w}}$ that satisfies the constraints, i.e., $\mathbf{C}^{\mathrm{T}}\mathbf{w} = \mathbf{C}^{\mathrm{T}}\mathbf{F} = \mathbf{f}$ [6]. Vector $\mathbf{w}_{GSC}$ is not affected by the constraints and may be adapted using unconstrained adaptation algorithms in order to reduce interference from signal components that lie within the null space of $\mathbf{C}$.

It is clear from the previous discussion that coefficient adaptation for the GSC model is performed within a reduced-dimension subspace. The transformation $\mathbf{T}$ in (13) applied onto the input-signal vector is such that the lower part of this transformed input $\mathbf{B}^{\mathrm{T}}\mathbf{x}(k)$ is restricted to the null space of the constraint matrix $\mathbf{C}$, which is of dimension $MN-p$. Therefore, adaptation along $\mathbf{B}^{\mathrm{T}}\mathbf{x}(k)$ causes no departure from the constraint hyperplane. Another important factor to stress is that the input signal of the adaptive filter is persistently exciting of order $MN-p$.

The structure of matrix $\mathbf{B}$ plays an important role in the GSC structure for its choice determines the computational complexity and, in many cases, the robustness against numerical instabilities of the overall system [12]. If SVD or any other decomposition is employed, the resulting nonsquared ($MN \times MN-p$) matrix $\mathbf{B}$ will, in general, have no exploitable special structure. This may result in a highly inefficient implementation with computational complexity up to one order of magnitude higher than that of the adaptation algorithm itself. This is due to the matrix-vector multiplication $\mathbf{Bx}(k)$ performed at each iteration. Although, in some applications [5], it may be possible to construct trivial blocking matrices whose elements are either 0, 1, or -1, these matrices pose some practical problems that may prevent their use. For instance, if matrix $\mathbf{B}$ is such that the transformation matrix $\mathbf{T}$ is not unitary, then the transients of the adaptive filters in the GSC model and in the LCMV may bear no relation [5]. Furthermore, if applied to the multistage Wiener filter structure presented in [11], nonunitary transformations invariably yield severe problems related to finite precision arithmetic [12]. The Householder decomposition used as suggested herein allows efficient implementation and results in a unitary transformation matrix. If necessary, the Householder reflections can be performed via dedicated CORDIC hardware or software [13].

III. Householder-Transform Constrained Algorithms

For a general constrained minimization problem, the multiplication of the blocking matrix by the input signal vector in a GSC structure may be computationally intensive and, for many applications, not practical. In this section, we will propose an elegant solution to this problem. The derivation of the first algorithm presented in this section starts from the CLMS algorithm [see (8)] and performs a rotation on vector $\mathbf{Px}(k)$ in order to make sure that the coefficient vector is never perturbed in a direction not excited by $\mathbf{Px}(k)$. This can be done if an orthogonal rotation matrix $\mathbf{Q}$ is used as the transformation that will generate a modified coefficient vector $\overline{\mathbf{w}}(k)$ that relates to $\mathbf{w}(k)$ according to

$$\overline{\mathbf{w}}(k) = \mathbf{Q}\mathbf{w}(k). \tag{19}$$

We can visualize this operation in Fig. 2 if we imagine axes $\mathbf{w}_1$ and $\mathbf{w}_2$ rotated counterclockwise by an angle θ.

If we choose the matrix $\mathbf{Q}$ such that $\mathbf{QQ}^{\mathrm{T}} = \mathbf{Q}^{\mathrm{T}}\mathbf{Q} = \mathbf{I}$ and

$$\overline{\mathbf{C}}(\overline{\mathbf{C}}^{\mathrm{T}}\overline{\mathbf{C}})^{-1}\overline{\mathbf{C}}^{\mathrm{T}} = \begin{bmatrix} \mathbf{I}_{p\times p} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{bmatrix} \tag{20}$$

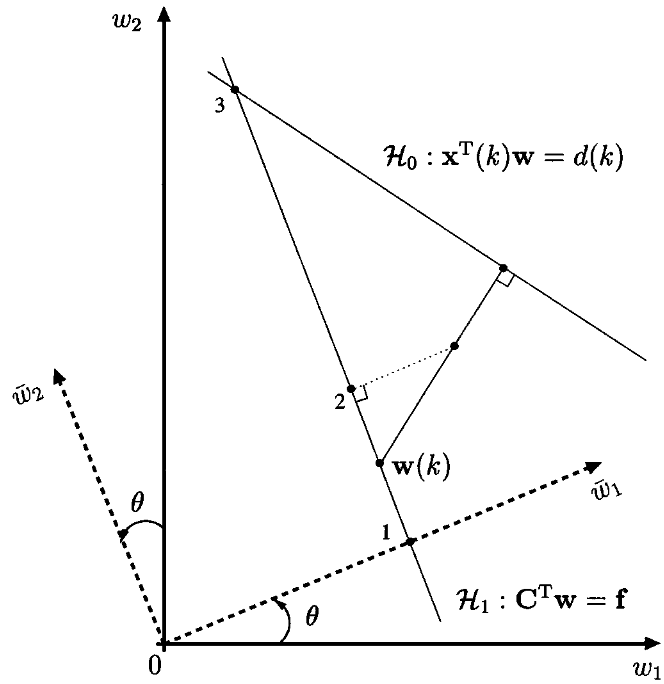

Fig. 4. Coefficient-vector rotation. 1) $\overline{\mathbf{w}}(0) = \mathbf{QF} = \mathbf{QC}(\mathbf{C}^{\mathrm{T}}\mathbf{C})^{-1}\mathbf{f}$. 2) $\mathbf{w}(k+1)$ for the HCLMS algorithm. 3) $\mathbf{w}(k+1)$ for the HNCLMS algorithm.

TABLE I
HCLMS Algorithm

Available at time instant k:
$\mathbf{x}(k)$, $\mathbf{C}$, $\mathbf{f}$, $\mathbf{Q}$, and μ (step-size)
Initialize:
$\bar{\mathbf{w}}(0) = \mathbf{QC}(\mathbf{C}^{\mathrm{T}}\mathbf{C})^{-1}\mathbf{f}$;
for $k = 0, 1, 2, \ldots$
{
$\bar{\mathbf{x}}(k) = \mathbf{Qx}(k)$;
$\bar{\mathbf{x}}_L(k) = MN-p$ last elements of $\bar{\mathbf{x}}(k)$;
$\bar{\mathbf{w}}(k) = \begin{bmatrix} \bar{\mathbf{w}}_U(0) \\ \bar{\mathbf{w}}_L(k) \end{bmatrix}$;
$y(k) = \bar{\mathbf{w}}^{\mathrm{T}}(k)\bar{\mathbf{x}}(k)$;
$\bar{\mathbf{w}}_L(k+1) = \bar{\mathbf{w}}_L(k) - \mu y(k)\bar{\mathbf{x}}_L(k)$;
}

then $\overline{\mathbf{C}} = \mathbf{QC}$ satisfies $\mathbf{f} = \overline{\mathbf{C}}^{\mathrm{T}}\overline{\mathbf{w}}(k+1)$, and the transformed projection matrix is such that

$$\overline{\mathbf{P}} = \mathbf{QPQ}^{\mathrm{T}} = \mathbf{I} - \overline{\mathbf{C}}(\overline{\mathbf{C}}^{\mathrm{T}}\overline{\mathbf{C}})^{-1}\overline{\mathbf{C}}^{\mathrm{T}} = \begin{bmatrix} \mathbf{0}_{p\times p} & \mathbf{0} \\ \mathbf{0} & \mathbf{I} \end{bmatrix}. \tag{21}$$

If $\overline{\mathbf{w}}(0)$ is initialized as

$$\overline{\mathbf{w}}(0) = \overline{\mathbf{C}}(\overline{\mathbf{C}}^{\mathrm{T}}\overline{\mathbf{C}})^{-1}\mathbf{f} = \mathbf{QF} \tag{22}$$

then its first p elements $\overline{\mathbf{w}}_U(0)$ need not be updated. The update equation of the proposed algorithm, named the

TABLE II
COMPUTATION OF $\bar{\mathbf{x}}_k = \mathbf{Q}\mathbf{x}_k$

$$
\begin{aligned}
&\bar{\mathbf{x}}_k = \mathbf{x}(k);\\
&\text{for } i = 1:p\\
&\{\\
&\quad \bar{\mathbf{x}}_k(i:MN) = \bar{\mathbf{x}}_k(i:MN)\\
&\qquad -2\mathbf{V}(i:MN,i)\left[\mathbf{V}^{\mathrm{T}}(i:MN,i)\bar{\mathbf{x}}_k(i:MN)\right];\\
&\}\\
&\bar{\mathbf{x}}(k) = \bar{\mathbf{x}}_k;
\end{aligned}
$$

TABLE III
CONSTRUCTION OF MATRIX $\mathbf{V}$ CONTAINING THE HOUSEHOLDER VECTORS

$$
\begin{aligned}
&\text{Available at start:}\\
&\quad \mathbf{A} \text{ is an } MN\times p \text{ matrix to be triangularized}\\
&\text{Initialize:}\\
&\quad \mathbf{V} = 0_{MN\times p};\\
&\\
&\text{for } i = 1:p\\
&\{\\
&\quad \mathbf{x} = \mathbf{A}(i:MN,i);\\
&\quad \mathbf{e_1} = [\mathbf{1}\ \ \mathbf{0_{1\times(MN-i)}}]^{\mathrm{T}}\\
&\quad \mathbf{v} = \text{sign}\,(\mathbf{x}(1))\,\|\mathbf{x}\|\mathbf{e_1} + \mathbf{x};\\
&\quad \mathbf{v} = \mathbf{v}/\|\mathbf{v}\|;\\
&\quad \mathbf{A}(i:MN,i:p) = \mathbf{A}(i:MN,i:p) - 2\mathbf{v}\left(\mathbf{v}^{\mathrm{T}}\mathbf{A}(i:MN,i:p)\right);\\
&\quad \mathbf{V}(i:MN,i) = \mathbf{v};\\
&\}
\end{aligned}
$$

Householder-transform constrained LMS [14], is obtained by premultiplying (8) by $\mathbf{Q}$

$$
\begin{aligned}
\overline{\mathbf{w}}&(k+1)\\
&= \mathbf{Q}\mathbf{w}(k+1) = \mathbf{Q}\left\{\mathbf{P}[\mathbf{w}(k) - \mu y(k)\mathbf{x}(k)] + \mathbf{F}\right\}\\
&= \left[\mathbf{Q}\mathbf{P}\mathbf{Q}^{\mathrm{T}}\right][\mathbf{Q}\mathbf{w}(k)] - \mu y(k)\left[\mathbf{Q}\mathbf{P}\mathbf{Q}^{\mathrm{T}}\right][\mathbf{Q}\mathbf{x}(k)] + \mathbf{Q}\mathbf{F}\\
&= \begin{bmatrix} \mathbf{0}_{p\times p} & \mathbf{0}\\ \mathbf{0} & \mathbf{I}\end{bmatrix}\overline{\mathbf{w}}(k) - \mu y(k)\begin{bmatrix} \mathbf{0}_{p\times p} & \mathbf{0}\\ \mathbf{0} & \mathbf{I}\end{bmatrix}\overline{\mathbf{x}}(k) + \begin{bmatrix}\overline{\mathbf{w}}_U(0)\\ \mathbf{0}\end{bmatrix}\\
&= \begin{bmatrix}\overline{\mathbf{w}}_U(0)\\ \overline{\mathbf{w}}_L(k+1)\end{bmatrix} = \begin{bmatrix}\overline{\mathbf{w}}_U(0)\\ \overline{\mathbf{w}}_L(k)\end{bmatrix} - \mu y(k)\begin{bmatrix}\mathbf{0}\\ \overline{\mathbf{x}}_L(k)\end{bmatrix}
\end{aligned}\tag{23}
$$

where $\overline{\mathbf{w}}_L(k)$ and $\overline{\mathbf{x}}_L(k)$ denote the $MN - p$ last elements of vectors $\overline{\mathbf{w}}(k)$ and $\overline{\mathbf{x}}(k)$, respectively. Note that vector $\overline{\mathbf{C}}(\overline{\mathbf{C}}^{\mathrm{T}}\overline{\mathbf{C}})^{-1}\mathbf{f}$ has only p nonzero elements.

Although the solution $\overline{\mathbf{w}}(k)$ is biased by a transformation $\mathbf{Q}$, the output signal and, consequently, the output error are not modified by the transformation. We conclude, therefore, that the proposed algorithm minimizes the same objective function minimized by the CLMS algorithm.

A. Choice of the Transformation Matrix $\mathbf{Q}$

We maintain that matrix $\mathbf{Q}$ in (19) may be constructed with successive Householder transformations [15] applied onto each of the p columns of matrix $\mathbf{CL}$, where $\mathbf{L}$ is the square-root factor of matrix $(\mathbf{C}^{\mathrm{T}}\mathbf{C})^{-1}$, i.e., $\mathbf{L}\mathbf{L}^{\mathrm{T}} = (\mathbf{C}^{\mathrm{T}}\mathbf{C})^{-1}$.

Theorem 1: If

$$\mathbf{Q} = \mathbf{Q}_p \cdots \mathbf{Q}_2\mathbf{Q}_1 \tag{24}$$

where

$$\mathbf{Q}_i = \begin{bmatrix}\mathbf{I}_{i-1\times i-1} & \mathbf{0}^{\mathrm{T}}\\ \mathbf{0} & \overline{\mathbf{Q}}_i\end{bmatrix} \tag{25}$$

and $\overline{\mathbf{Q}}_i$ is an $(MN - i + 1)\times(MN - i + 1)$ Householder transformation matrix on the form $\overline{\mathbf{Q}}_i = \mathbf{I} - 2\overline{\mathbf{v}}_i\overline{\mathbf{v}}_i^{\mathrm{T}}$ [15], then (20) is satisfied.

Proof: After $i - 1$ transformations, matrix $\mathbf{Q}_{i-1}\cdots\mathbf{Q}_1\mathbf{CL}$ may be partitioned as

$$
\mathbf{Q}_{i-1}\cdots\mathbf{Q}_1\mathbf{CL} = \left[\begin{array}{c:c} \overbrace{\mathbf{D}^{(i)}}^{i-1} & \overbrace{\mathbf{E}^{(i)}}^{MN-i+1}\\ \hdashline \mathbf{0} & \mathbf{A}^{(i)}\end{array}\right]\begin{array}{l}\}\, i-1\\ \\ \}\, MN-i+1.\end{array}\tag{26}
$$

The $i-1$ Householder transformations make matrix $\mathbf{D}^{(i)}$ upper triangular. Now, let $\mathbf{a}_j^{(i)}$ denote the jth column of the $\mathbf{A}^{(i)}$ matrix. It is easy to show by carrying out the Householder transformation $\overline{\mathbf{Q}}_i$ on $\mathbf{A}^{(i)}$ that if the columns, *viz.* $\{\mathbf{a}_j^{(i)},\ \ j = 1,\ \ldots,\ MN - i + 1\}$, of $\mathbf{A}^{(i)}$ satisfy

$$\left\|\mathbf{a}_j^{(i)}\right\| = 1 \tag{27}$$

$$\left(\mathbf{a}_i^{(i)}\right)^{\mathrm{T}}\mathbf{a}_j^{(i)} = \delta_{ij} \tag{28}$$

then

$$\overline{\mathbf{Q}}_i\mathbf{A}^{(i)} = \begin{bmatrix}\pm 1 & 0 & \cdots & 0\\ 0 & \star & \cdots & \star\\ \vdots & \vdots & \ddots & \vdots\\ 0 & \star & \cdots & \star\end{bmatrix}. \tag{29}$$

For $i = 1$, $\overline{\mathbf{Q}}_1 = \mathbf{Q}_1$, and $\mathbf{A}^{(1)} = \mathbf{CL}$. Matrix $\mathbf{CL}$ has orthonormal columns because $(\mathbf{CL})^{\mathrm{T}}\mathbf{CL} = \mathbf{L}^{\mathrm{T}}(\mathbf{L}\mathbf{L}^{\mathrm{T}})^{-1}\mathbf{L} = \mathbf{I}$. Therefore, (27) and (28) are directly satisfied. By the *fundamental theorem of inner product invariance in Householder transforms* [16], orthonormality is maintained for $\mathbf{Q}_1\mathbf{CL}$ and, by induction, (27) and (28) are also satisfied for any $i > 1$. As a consequence, $\mathbf{D}^{(i)}$ is a diagonal matrix with ± 1 entries, and $\mathbf{E}^{(i)}$ is a matrix of zeros. This concludes the proof. □

Notice that the ± 1 entries in matrix $\mathbf{D}^{(i)}$ result from the robust implementation of the Householder transformation given in [15].

From (23), we verify that the algorithm updates the coefficients in a subspace with reduced dimension. The entries of vector $\mathbf{w}(k)$ that lie in the subspace of the constraints need not be updated. Due to the equivalence of Householder reflections and Givens rotations [17], a succession of Givens rotations could also be used. However, rotations are not as efficiently implemented as reflections and computational complexity might render the resulting algorithm not practical.

B. Normalized HCLMS Algorithm

A normalized version of the HCLMS algorithm, namely, the NHCLMS algorithm [14], can be derived, and its update equation is

$$
\begin{aligned}
\overline{\mathbf{w}}(k+1) &= \begin{bmatrix}\overline{\mathbf{w}}_U(0)\\ \overline{\mathbf{w}}_L(k+1)\end{bmatrix}\\
&= \begin{bmatrix}\overline{\mathbf{w}}_U(0)\\ \overline{\mathbf{w}}_L(k)\end{bmatrix} - \mu\frac{y(k)}{\overline{\mathbf{x}}_L^{\mathrm{T}}(k)\overline{\mathbf{x}}_L(k)}\begin{bmatrix}\mathbf{0}\\ \overline{\mathbf{x}}_L(k)\end{bmatrix}.
\end{aligned}\tag{30}
$$

TABLE IV
COMPUTATIONAL COMPLEXITY

ALGORITHM	ADDITIONS	MULTIPLICATIONS	DIVISIONS
CLMS	$(2p+2)MN-(p+1)$	$(2p+2)MN+1$	0
NCLMS	$(3p+3)MN-(p+2)$	$(3p+3)MN+1$	1
GSC-LMS	$(MN)^2+(2-p)MN-(1+p)$	$(MN)^2+(3-p)MN-(2p-1)$	0
GSC-NLMS	$(MN)^2+(3-p)MN-2(1+p)$	$(MN)^2+(4-p)MN-(3p-1)$	1
GSC-LMS with $\mathbf{B}$ of [18]	$(3+p)MN-\frac{p(p+5)}{2}-1$	$(3+2p)MN-p(p+3)+1$	0
GSC-NLMS with $\mathbf{B}$ of [18]	$(4+p)MN-\frac{p(p+7)}{2}-2$	$(4+2p)MN-p(p+4)+1$	1
HCLMS	$(2p+2)MN-(p^2+p+1)$	$(2p+2)MN-(p^2-1)$	0
NHCLMS	$(2p+3)MN-(p^2+2p+2)$	$(2p+3)MN-(p^2+p-1)$	1

Note that the Householder transformation allows normalization without the need of multiplication by a projection matrix, as it is required for the NCLMS in (11).

Fig. 4 illustrates the coefficient update for the HCLMS and the NHCLMS algorithms. Note, in this figure, that a rotation by θ is performed on the coordinate system $\overline{\mathbf{w}} = [\overline{w}_1\, \overline{w}_2]^{\mathrm{T}} = \mathbf{Q}\mathbf{w}(k) = \mathbf{Q}[w_1\ w_2]^{\mathrm{T}}$. This angle is chosen such that the rotated axis $\overline{w}_2$ becomes parallel to the constraint hyperplane, and the coordinate corresponding to $\overline{w}_1$ needs no further update. This is so because $\overline{w}_1$ becomes orthogonal to $\mathcal{H}_1$. Table I shows an algorithmic description of the HCLMS algorithm.

C. Computational Complexity Issues

In this section, we explain why and how the implementation via Householder transformation is better than the GSC and the constrained alternatives. Let us start with the procedure used to compute the product $\mathbf{Q}\mathbf{x}(k)$. In order to have an efficient Householder implementation, the transformation of the input-signal vector in every iteration is carried out through p reflections given by

$$\overline{\mathbf{x}}(k) = \mathbf{Q}\mathbf{x}(k) = \mathbf{Q}_p \cdots \mathbf{Q}_2\mathbf{Q}_1\mathbf{x}(k) \tag{31}$$

where

$$\mathbf{Q}_i = \begin{bmatrix} \mathbf{I}_{i-1\times i-1} & \mathbf{0}^{\mathrm{T}} \\ \mathbf{0} & \overline{\mathbf{Q}}_i \end{bmatrix} \tag{32}$$

and matrix $\overline{\mathbf{Q}}_i = \mathbf{I} - 2\overline{\mathbf{v}}_i\overline{\mathbf{v}}_i^{\mathrm{T}}$ is a $(MN-i+1)\times(MN-i+1)$ Householder transformation matrix [15].

If we define the vector $\mathbf{v}_i^{\mathrm{T}} = [\mathbf{0}_{i-1}^{\mathrm{T}}\ \overline{\mathbf{v}}_i^{\mathrm{T}}]^{\mathrm{T}}$, where the $p\times 1$ vector $\mathbf{0}_{i-1}$ introduces $i-1$ zeros before $\overline{\mathbf{v}}_i$, we can construct matrix $\mathbf{V} = [\mathbf{v}_1\ \mathbf{v}_2\ \cdots\ \mathbf{v}_p]$, and the factored product in (31) could be implemented with the procedure described in Table II. Furthermore, the procedure for the calculation of the Householder vectors and the resulting $\mathbf{V}$ is described in Table III. In Table III, $\mathbf{A}$ is the matrix to be triangularized, which, in our particular case of interest, corresponds to $\mathbf{A} = \mathbf{C}\mathbf{L}$, where $\mathbf{L}$ is the square-root factor of the matrix $(\mathbf{C}^{\mathrm{T}}\mathbf{C})^{-1}$, as proposed earlier in this section.

From Table II, we see that the computation of $\overline{\mathbf{x}}(k) = \mathbf{Q}\mathbf{x}_k$ using the product representation in (31) only involves $2MNp - p(p-1)$ multiplications and $2MNp - p^2$ additions. Table IV shows the computational complexity for the CLMS, NCLMS, HCLMS, and NHCLMS algorithms and the GSC implementation of the CLMS and NCLMS algorithms. The computational complexity for the GSC implementation is given for two choices of the blocking matrix $\mathbf{B}$. The first implementation uses a matrix $\mathbf{B}$ obtained by, e.g., SVD leading to an inefficient implementation of the multiplication $\mathbf{B}\mathbf{x}(k)$. The second implementation, which is applicable only in certain problems, uses a matrix $\mathbf{B}$ constructed through a cascade of sparse matrices as presented in [18], rendering an implementation of the multiplication $\mathbf{B}\mathbf{x}(k)$ of low computational complexity.

D. Householder-Transform Constrained Algorithms and the GSC

Fig. 5 shows, step-by-step, the relation between a Householder-constrained (HC) algorithm and the GSC structure. If $\mathbf{Q}$ is factored into an upper part and lower part (see Fig. 5), it is easy to show that $\mathbf{Q}_L$ spans the null space of $\mathbf{C}$ and may be regarded as a valid blocking matrix. Furthermore, $\mathbf{Q}_U^{\mathrm{T}}\overline{\mathbf{w}}_U(0) = \mathbf{C}(\mathbf{C}^{\mathrm{T}}\mathbf{C})^{-1}\mathbf{f} = \mathbf{F}$, which is the upper part of the GSC structure. However, we stress that for most practical values of p, the implementation of $\mathbf{Q}_L$ and $\mathbf{Q}_U$ separately renders much higher computational complexity because it does not take advantage of the efficiency of the Householder transformation. The transformed input-signal vector can be efficiently obtained via p Householder transformations, which require only p inner products. We maintain that our approach can be regarded as a GSC structure and, therefore, any unconstrained adaptive algorithm can be used to update $\mathbf{w}_L(k)$. As an example of this assertion, Table V shows the equations of the Householder-transform-constrained quasi-Newton (HCQN) algorithm obtained directly from [10] and Fig. 5, as previously reported in [19]. Notice that the algorithm in Table V does not require the inversion and construction of the $p\times p$ matrix encountered in the conventional CQN algorithm presented in [10], resulting in a much simpler implementation of the algorithm.

IV. SIMULATION RESULTS

In this section, the performance of the proposed algorithms is evaluated through simulations and compared with their GSC counterparts.

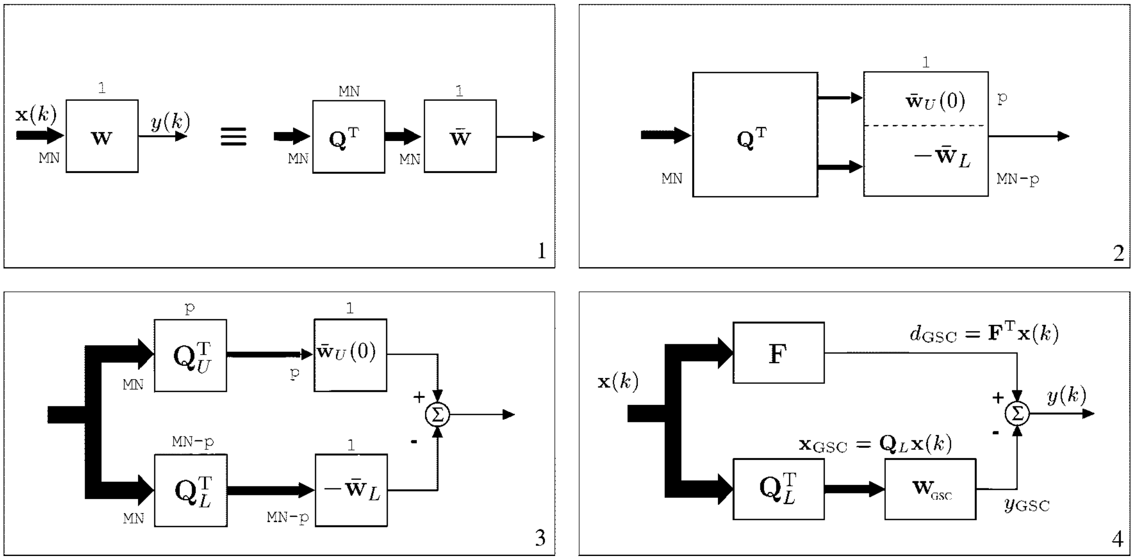

Fig. 5. HC adaptive filter under the GSC model. 1) Applying the transformation as in (19). 2) Splitting the transformed vector as in (23). 3) Partitioning Q in order to reach the GSC equivalent. 4) HC algorithm under a GSC perspective.

TABLE V
HCQN ALGORITHM

Available at time instant k:
  $\mathbf{x}(k)$, $\mathbf{C}$, $\mathbf{f}$, and $\mathbf{Q}$
Initialize:
  α, $\bar{\mathbf{R}}_L^{-1}(0)$, and $\bar{\mathbf{w}}(0) = \mathbf{QC}(\mathbf{C}^{\mathrm{T}}\mathbf{C})^{-1}\mathbf{f}$

for $k = 1, 2, \ldots$
{
  $\bar{\mathbf{x}}(k) = \mathbf{Q}\mathbf{x}(k)$;
  $\bar{\mathbf{x}}_U(k) = p$ first elements of $\bar{\mathbf{x}}(k)$;
  $\bar{\mathbf{x}}_L(k) = MN - p$ last elements of $\bar{\mathbf{x}}(k)$;
  $\bar{\mathbf{w}}(k-1) = \begin{bmatrix} \bar{\mathbf{w}}_U(0) \\ -\bar{\mathbf{w}}_L(k-1) \end{bmatrix}$;
  $\bar{e}(k) = \bar{\mathbf{w}}_U^{\mathrm{T}}(0)\bar{\mathbf{x}}_U(k) - \bar{\mathbf{w}}_L^{\mathrm{T}}(k-1)\bar{\mathbf{x}}_L(k)$;
  % $\bar{e}(k)$ is equivalent to the *a priori* output
  % or $y(k) = \bar{\mathbf{w}}^{\mathrm{T}}(k-1)\bar{\mathbf{x}}(k)$
  $\mathbf{t}(k) = \bar{\mathbf{R}}_L^{-1}(k-1)\bar{\mathbf{x}}_L(k)$;
  $\tau = \bar{\mathbf{x}}_L^{\mathrm{T}}(k)\mathbf{t}(k)$;
  $\mu(k) = \frac{1}{2\tau(k)}$;
  $\bar{\mathbf{R}}_L^{-1}(k) = \bar{\mathbf{R}}_L^{-1}(k-1) + \frac{\mu(k)-1}{\tau(k)}\mathbf{t}(k)\mathbf{t}^{\mathrm{T}}(k)$;
  $\bar{\mathbf{w}}_L(k) = \bar{\mathbf{w}}_L(k-1) + \alpha\frac{\bar{e}(k)}{\tau(k)}\mathbf{t}(k)$;
}

A uniform linear array with $M = 12$ antennas with element spacing equal to a half wavelength was used in a system with $K = 5$ users, where the signal of one user is of interest, and the other four signals are treated as interferers. The direction of arrival (DOA) and the signal-to-noise ratio (SNR) for the different signals can be found in Table VI.

A second-order derivative constraint matrix [20] was used, giving a total of three constraints. For the GSC implementation, the nonunitary blocking matrix in [18] was used.

TABLE VI
SIGNAL PARAMETERS

SIGNAL	DOA	SNR
desired	0°	15 dB
interferer 1	22°	20 dB
interferer 2	-15°	25 dB
interferer 3	-20°	25 dB
interferer 4	-50°	20 dB

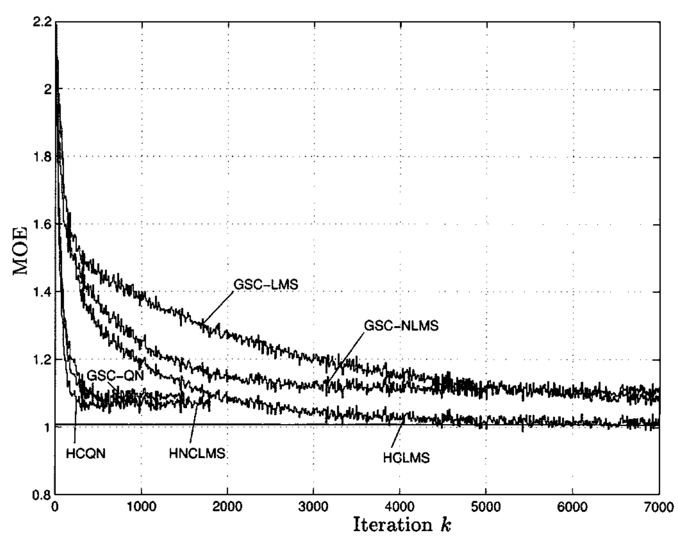

Fig. 6. Learning curves of the algorithms.

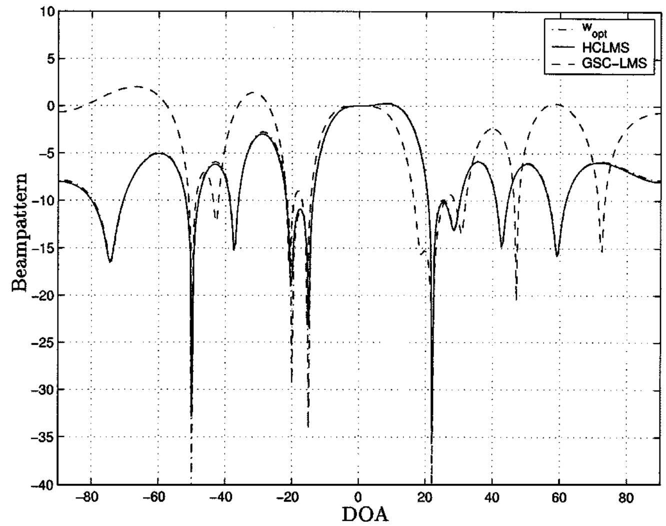

Fig. 7. Beampattern for the HCLMS and GSC-LMS algorithms.

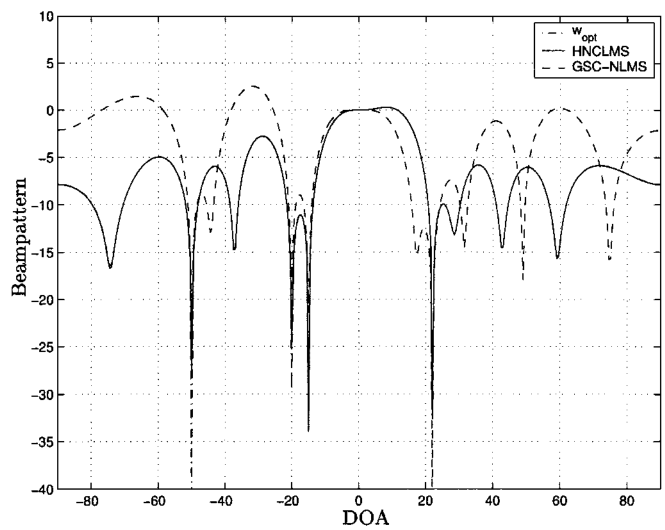

Fig. 8. Beampattern for the HNCLMS and the GSC-NLMS algorithms.

Fig. 6 shows the learning curves of the different algorithms. The results were obtained by averaging 2000 realizations of the experiment. The step sizes used in the algorithms were

- $\mu = 5 \cdot 10^{-4}$ for the CLMS and the HCLMS algorithms;
- $\mu = 10^{-5}$ for the GSC-LMS algorithm;
- $\mu_n = 0.05$ for the NLMS algorithms;
- $\alpha = 0.05$ for the QN algorithms.

As can be seen from Fig. 6, the Householder implementations have a better performance than the corresponding GSC implementations using the sparse blocking matrix.

Figs. 7–9 show the beampatterns resulting from the different algorithms. The beampatterns obtained with the Householder algorithms are very close to the optimal solution. On the other hand, the GSC-based implementations failed to suppress completely all interferers at the same time, which suggests that the adaptation algorithms did not achieve a steady state, even after 7000 iterations. The output gains in the directions of the interferers are shown in Table VII.

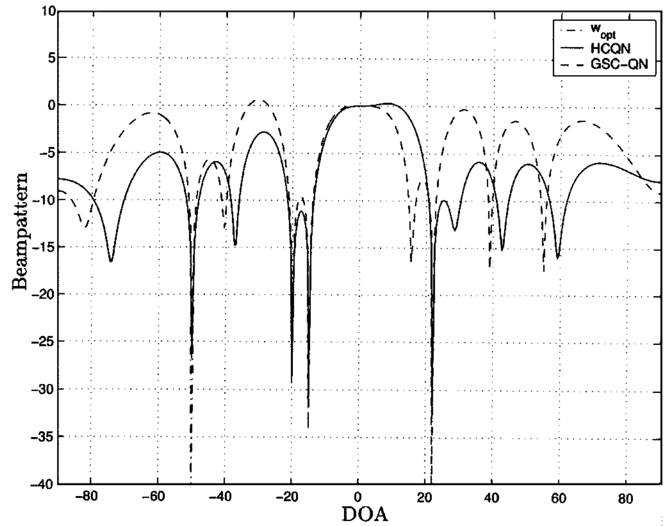

Fig. 9. Beampattern for the HCQN and the GSC-QN algorithms.

TABLE VII
OUTPUT GAINS IN THE DIRECTIONS OF THE INTERFERERS

ALGORITHM	$\theta = 22^o$	$\theta = -15^o$	$\theta = -20^o$	$\theta = -50^o$
GSC-LMS	-26.20 dB	-13.29 dB	-11.76 dB	-22.99 dB
GSC-NLMS	-31.74 dB	-21.39 dB	-18.73 dB	-22.67 dB
GSC-QN	-26.97 dB	-27.29 dB	-22.79 dB	-24.88 dB
HCLMS	-35.40 dB	-23.38 dB	-17.94 dB	-32.85 dB
NHCLMS	-31.50 dB	-33.85 dB	-24.82 dB	-27.83 dB
HCQN	-32.92 dB	-30.76 dB	-28.66 dB	-26.75 dB

V. CONCLUSIONS

In this paper, we presented an efficient implementation of linearly-constrained minimum-variance adaptive filters based on the Householder transformation of the input signal. With this type of transformation, we derived several adaptation algorithms for LCMV applications, such as the Householder-transform constrained least mean square algorithm and its normalized version, and maintained that extension to other adaptation algorithms should be trivial.

Via the Householder transformation, we were able to reduce the dimension of the subspace in which the adaptive-filter coefficients are updated, therefore obtaining a transformed input signal that is persistently exciting. Viewed under the perspective of the generalized sidelobe canceling model, we showed that the transformation matrix can be factored into a matrix satisfying the constraints and a blocking matrix.

In terms of computational complexity, our method is comparable with the most efficient implementations of the blocking matrix found in the literature, with the advantage that the Householder transformation, and, consequently, the blocking matrix implicitly used in the transformation, are unitary. For this reason, not only the steady-state mean squared output error is the same as that of the conventional nontransformed LCMV filter, but the equivalence is also verified during the transient.

Having a unitary transformation also imparts robustness to the method [12], for example, when applied to nonconventional Wiener-filter structures (e.g., multistage representation). Some of these properties were illustrated in one example of beamforming.

REFERENCES

[1] B. Widrow, P. E. Mantey, L. J. Griffiths, and B. B. Goode, "Adaptive antenna systems," *Proc. IEEE*, vol. 55, pp. 2143–2159, Dec. 1967.

[2] L. J. Griffiths, "A simple adaptive algorithm for real-time processing in antenna arrays," *Proc. IEEE*, vol. 57, pp. 1696–1704, Oct. 1969.

[3] O. L. Frost, III, "An algorithm for linearly constrained adaptive array processing," *Proc. IEEE*, vol. 60, pp. 926–935, Aug. 1972.

[4] G. C. Goodwin and S. K. Sin, *Adaptive Filtering Prediction and Control*. Englewood Cliffs, NJ: Prentice-Hall, 1984.

[5] L. J. Griffiths and C. W. Jim, "An alternative approach to linearly constrained adaptive beamforming," *IEEE Trans. Antennas Propagat.*, vol. AP-30, pp. 27–34, Jan. 1982.

[6] S. Haykin, *Adaptive Filter Theory*, 3rd ed. Englewood Cliffs, NJ: Prentice-Hall, 1996.

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

[8] S. C. Park and J. F. Doherty, "Generalized projection algorithm for blind interference suppression in DS/CDMA communications," *IEEE Trans. Circuits Syst. II*, vol. 44, pp. 453–460, June 1997.

[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 Processing*, vol. 44, pp. 1168–1174, May 1996.

[10] M. L. R. de Campos, S. Werner, J. A. Apolinário, Jr., and T. I. Laakso, "Constrained quasi-Newton algorithm for CDMA mobile communications," in *Proc. Int. Telecommun. Symp.*, São Paulo, Brazil, Aug. 1998, pp. 371–376.

[11] J. S. Goldstein, I. S. Reed, and L. L. Scharf, "A multistage representation of the Wiener filter based on orthogonal projections," *IEEE Trans. Inform. Theory*, vol. 44, pp. 2943–2959, Nov. 1998.

[12] M. Honig and M. K. Tsatsanis, "Adaptive techniques for multiuser CDMA receivers," *IEEE Signal Processing Mag.*, pp. 49–61, May 2000.

[13] S.-F. Hsiao and J.-M. Delosme, "Householder CORDIC algorithms," *IEEE Trans. Comput.*, vol. 43, pp. 331–336, Jan. 1995.

[14] M. L. R. de Campos, S. Werner, and J. A. Apolinário, Jr., "Householder-transform constrained LMS algorithm with reduced-rank updating," in *Proc. Int. Conf. Acoust., Speech, Signal Process.*, vol. IV, Phoenix, AZ, Mar. 1999, pp. 1857–1860.

[15] G. H. Golub and C. F. Van Loan, *Matrix Computations*. Baltimore, MD: Johns Hopkins Univ. Press, 1983.

[16] A. O. Steinhardt, "Householder transforms in signal processing," *IEEE Acoust., Speech, Signal Processing Mag.*, vol. 5, pp. 4–12, July 1988.

[17] J. H. Wilkinson, *The Algebraic Eigenvalue Problem*. Oxford, U.K.: Oxford Univ. Press, 1965.

[18] C.-Y. Tseng and L. J. Griffiths, "A systematic procedure for implementing the blocking matrix in decomposed form," in *Proc. Twenty-Second Asilomar Conf. Signals, Syst., Comput.*, vol. 2, Pacific Grove, CA, 1988, pp. 808–812.

[19] C. A. Medina, C. V. Rodríguez, and J. A. Apolinário, Jr., "On the use of householder transformation in adaptive microphone array," in *Proc. 4th World Multiconf. Syst., Cybern., Inform.*, Orlando, FL, July 2000.

[20] Y. Chu and W.-H. Fang, "A novel wavevelet-based generalized sidelobe canceller," *IEEE Trans. Antennas Propagat.*, vol. 47, pp. 1485–1494, Sept. 1999.

Marcello L. R. de Campos (S'89–M'96) was born in Niterói, 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 a post-doctoral fellow with the Department of Electronics, School of Engineering, UFRJ, and the Program of Electrical Engineering, COPPE/UFRJ. From January 1997 until May 1998, he was an Associate Professor with the Department of Electrical Engineering (DE/3), Military Institute of Engineering (IME), Rio de Janeiro. He is currently Associate Professor with 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. 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. In 2001, he received a Nokia Visiting Fellowship to visit the University of Oulu, Oulu, Finland.

Stefan Werner (S'98) received the M.Sc. degree in electrical engineering from the Royal Institute of Technology (KTH), Stockholm, Sweden, in 1998. He is currently pursuing the Ph.D. degree at the Signal Processing Laboratory, Helsinki University of Technology, Espoo, Finland,

His research interests are in multiuser communications and adaptive filtering.

José Antonio Apolinário, Jr. (M'99) 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) 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 at the Helsinki University of Technology, Espoo, Finland, in 1997. His research interests comprise many aspects of 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 has been the Chapter Chair since 2001.