fits are possible if the Gaussians are less constrained, that is, if more of their characteristics (amplitude, position, and width) are used. This is demonstrated in Table I. For the first example, the position of the Gaussians was held at the extrema of $\sin(2\pi t)$, as done in [1], but the remaining parameters were adjusted to minimize the squared error. The result was a very good fit. Using all parameters, as in the second example, reduced the squared error even further, by four orders of magnitude.

As stated in [1], this is a nonlinear least squares curve fitting problem. Hence, we see the practical importance of suboptimal but efficient algorithms like the one proposed in that work.

References

[1] J. Ben-Arie and K. R. Rao, "Nonorthogonal signal representation by Gaussians and Gabor functions," *IEEE Trans. Circuits Syst. II*, vol. 42, pp. 402–413, June 1995.

[2] P. J. Davis, *Interpolation and Approximation.* Waltham, MA: Blaisdell, 1963 (reprinted by Dover, 1975).

[3] N. Wiener, *The Fourier Integral and Certain of Its Applications.* Cambridge, U.K.: Cambridge Univ. Press, 1933 (reprinted by Dover, 1958).

[4] K. Chandrasekharan, *Classical Fourier Transforms.* Berlin, Germany: Springer, 1989.

A Versatile Algorithm for Two-Dimensional Symmetric Noncausal Modeling

George-Othon Glentis, Cornelis H. Slump, and Otto E. Herrmann

***Abstract*—In this brief, a novel algorithm is presented for the efficient two-dimensional (2-D) symmetric noncausal finite impulse response (FIR) filtering and autoregressive (AR) modeling. Symmetric filter masks of general boundaries are allowed. The proposed algorithm offers the greatest maneuverability in the 2-D index space in a computationally efficient way. This flexibility can be taken advantage of if the shape of the 2-D mask is not *a priori* known and has to be dynamically configured.**

***Index Terms*—**

Algorithms, filtering, image processing, least mean square error methods, Toeplitz matrices.

I. Introduction

Two-dimensional least squares noncausal modeling is of great importance in a wide range of applications. These include image restoration, image enhancement, image compression, 2-D spectral estimation, detection of changes in image sequences, stochastic texture modeling, edge detection, etc. [1].

Let $x(n_1, n_2)$ be the input of a linear, space invariant 2-D FIR filter. The filter's output $y(n_1, n_2)$ is a linear combination of past input values $x(n_1 - i_1, n_2 - i_2)$ weighted by the *filter coefficients* c_{i_1,i_2} over a support region, or *filter mask* $\mathcal{M}$:

$$y(n_1, n_2) = -\sum_{(i_1,i_2)\in\mathcal{M}} c_{i_1,i_2} x(n_1 - i_1, n_2 - i_2). \quad (1)$$

A fairly general shape for the support region is considered. Thus, $\mathcal{M}$ is allowed to be horizontally convex, i.e., the horizontal line segment joining any two points $(i_1, i_2), (i_1, i_3) \in \mathcal{M}$ lies in $\mathcal{M}$.

The filter is restricted to be linear phase. Thus, the following conditions should be satisfied [1]:

$$\begin{aligned} &\text{mask symmetry} \quad \forall (i_1, i_2) \in \mathcal{M}, \quad \exists(-i_1, -i_2) \in \mathcal{M} \\ &\text{coeff. symmetry} \quad c_{i_1,i_2} = c_{-i_1,-i_2}. \end{aligned}$$

Given an input 2-D signal $x(n_1, n_2)$ and a desired response 2-D signal $z(n_1, n_2)$, the optimal mean-squared error (MSE) 2-D FIR filter is obtained by minimizing the cost function

$$\mathcal{E}[(z(n_1, n_2) - y(n_1, n_2))^2]. \quad (2)$$

$\mathcal{E}[\cdot]$ is the expectation operator. MSE 2-D linear prediction can be handled as a special case of filtering, setting $z(n_1, n_2) = x(n_1, n_2)$ and excluding the origin $\{(0, 0)\}$ from the filter mask, i.e., $(i_1, i_2) \in \mathcal{M} - \{(0, 0)\}$.

Minimization of (2) with respect to the filter parameters c_{i_1,i_2} leads to a system of linear system of equations, the so-called normal equations. Any well-behaved linear system solver can be applied for the inversion of the 2-D normal equations. However, the special structure of the normal equations gives rise to the development of cost-effective algorithms for the determination of the unknown parameters [2]–[4].

In this paper a new, highly efficient algorithm is developed for the solution of the normal equations in a ***true*** order recursive way [7]. Filter masks of general, horizontally convex shape are allowed. Fast recursions are developed for the updating of lower order filter parameters toward any neighboring point. It can efficiently be applied for the order-recursive estimation of the 2-D MSE FIR filter and system identification, accelerating the exhaustive search procedures required by most of the order determination criteria [8], [9].

II. 2-D Symmetric Support Region

Consider the support region depicted in Fig. 1. More precisely, $\mathcal{M}$ consists of a union of intervals:

$$\mathcal{M} = \bigcup_{i_1=-k_1}^{k_1} \boldsymbol{m}(i_1),$$
$$\boldsymbol{m}(i_1) = \{(i_1, i_2)\colon -k_2(-i_1) \le i_2 \le k_2(i_1)\}.$$

Clearly, $k_1 = \max\{i_1\colon (i_1, i_2) \in \mathcal{M}\}$, $k_2(i_1) = \max\{i_2\colon (i_1, i_2) \in \boldsymbol{m}(i_1)\}$. Then, (1) takes the form

$$y(n_1, n_2) = -\sum_{i_1=-k_1}^{k_1} \sum_{i_2=-k_2(-i_1)}^{k_2(i_1)} c_{i_1,i_2} x(n_1 - i_1, n_2 - i_2).$$

The above equation can be written as a linear regression:

$$y(n_1, n_2) = -\mathcal{X}_{\mathcal{M}}^t(n_1, n_2)\mathcal{C}_{\mathcal{M}} \quad (3)$$

where the regressor (data vector) and the filter coefficients vector are defined by (4), (5), and (5a).

The filter coefficients' symmetry implies that

$$\mathcal{C}_{\mathcal{M}} = \mathcal{J}\mathcal{C}_{\mathcal{M}} \quad (6)$$

Manuscript received May 4, 1995; revised April 18, 1997. The work of G.-O. Glentis was supported by the European Community, Human Capital Mobility, EUROCHIP Program. This paper was recommended by Associate Editor T. Hinamoto.

G. -O. Glentis is with the Department of Informatics, University of Athens, Panepistmiopolis, TYPA Blds, Athens 15784, Greece (e-mail: gglentis@di.uoa.gr).

C. H. Slump and O. E. Herrmann are with the Department of Electrical Engineering (BSC), University of Twente, 7500 AE Enschede, The Netherlands.

Publisher Item Identifier S 1057-7130(98)01642-5.

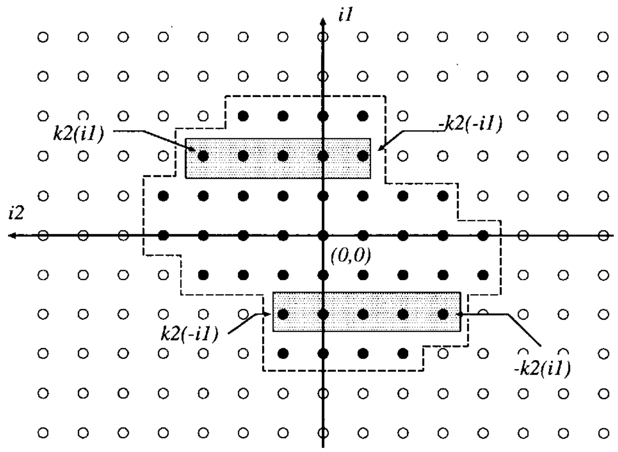

Fig. 1. Symmetric support region.

where $\mathcal{J}$ is a matrix with ones in the anti-diagonal, and zeros elsewhere. Clearly, $\mathcal{J}\mathcal{J} = \mathbf{I}$. Minimization of (2), subject to the symmetry constraint (6), leads to the following linear system of equations [3]:

$$(\mathcal{R}_{\mathcal{M}} + \mathcal{J}\mathcal{R}_{\mathcal{M}}\mathcal{J})\mathcal{C}_{\mathcal{M}} = -(\mathcal{D}_{\mathcal{M}} + \mathcal{J}\mathcal{D}_{\mathcal{M}}) \tag{7}$$

where $\mathcal{R}_{\mathcal{M}} = \mathcal{E}[\mathcal{X}_{\mathcal{M}}(n_1, n_2)\mathcal{X}_{\mathcal{M}}^t(n_1, n_2)]$ is the input signal autocorrelation matrix, and $\mathcal{D}_{\mathcal{M}} = \mathcal{E}[\mathcal{X}_{\mathcal{M}}(n_1, n_2)z(n_1, n_2)]$ is the cross-correlation vector between the input and the desired response signal.

In the sequel, real and homogeneous, random, wide-sense stationary 2-D signals will be considered. This implies that the autocorrelation between two samples depends on the difference of their coordinates [2]:

$$\begin{aligned}\mathcal{E}[x(n_1 - i_1, n_2 - i_2) \quad & x(n_1 - j_1, n_2 - j_2)] \\ &= \rho(i_1 - j_1, i_2 - j_2), \qquad \rho(i, j) = \rho(-i, -j).\end{aligned} \tag{8}$$

The autocorrelation matrix $\mathcal{R}_{\mathcal{M}}$ is a block matrix of block order $2k_1 + 1$

$$\mathcal{R}_{\mathcal{M}} = [\mathbf{R}(i_1, j_1)]_{\substack{i_1 = -k_1 \cdots k_1 \\ j_1 = -k_1 \cdots k_1}}$$

with entry Toeplitz matrices of the form

$$\begin{aligned}\mathbf{R}(i_1, j_1) &= \mathcal{E}\left[\mathbf{x}_{m(i_1)}(n_1, n_2)\mathbf{x}_{m(j_1)}^t(n_1, n_2)\right] \\ &= [\rho(i_1 - j_1, i_2 - j_2)]_{\substack{i_2 = -k_2(-i_1) \cdots k_2(i_1) \\ j_2 = -k_2(-j_1) \cdots k_2(j_1)}}.\end{aligned}$$

$\mathcal{D}_{\mathcal{M}}$ is a block vector $\mathcal{D}_{\mathcal{M}} = [\mathbf{d}(i_1)]_{i_1 = -k_1 \cdots k_1}$ with entry subvectors $\mathbf{d}(i_1) = [d(i_1, i_2)]_{i_2 = -k_2(-i_1) \cdots k_2(i_1)}$ where $d(i_1, i_2) = \mathcal{E}[x(n_1 - i_1, n_2 - i_2)z(n_1, n_2)]$.

In addition to the block Toeplitz structure, the autocorrelation matrix is perisymmetric, as follows from the symmetry of the support region and (8), i.e., $\mathcal{R}_{\mathcal{M}} = \mathcal{J}\mathcal{R}_{\mathcal{M}}\mathcal{J}$.

Thus, the normal equations (7) take the form

$$\mathcal{R}_{\mathcal{M}}\mathcal{C}_{\mathcal{M}} = -\mathcal{D}_{\mathcal{M}}^s \tag{9}$$

where $\mathcal{D}_{\mathcal{M}}^s = 1/2(\mathcal{D}_{\mathcal{M}} + \mathcal{J}\mathcal{D}_{\mathcal{M}})$.

III. The Algorithm

In this section, order-updating recursions are developed for the passage from lower order parameters to increased order counterparts. Single-step increments of the filter mask $\mathcal{M}$ are allowed each time. Thus, starting from $\mathcal{M}$, an increased order mask $\mathcal{M}^{i_1}$ is constructed with the addition of two symmetrically located neighboring samples.

Let us consider the increased order support region depicted in Fig. 2:

$$\mathcal{M}^{i_1} = \mathcal{M} \cup \{(i_1, k_2(i_1) + 1)\} \cup \{(-i_1, -k_2(i_1) - 1)\}.$$

The data vector associated with the increased order mask $\mathcal{M}^{i_1}$ is partitioned in such a way that the lower order data vector appears, i.e.,

$$\mathcal{X}_{\mathcal{M}^{i_1}}(n_1, n_2) = \mathcal{W}_{i_1}^t \begin{bmatrix} x(n_1 + i_1, n_2 + k_2(i_1) + 1) \\ \mathcal{X}_{\mathcal{M}}(n_1, n_2) \\ x(n_1 - i_1, n_2 - k_2(i_1) - 1) \end{bmatrix}. \tag{10}$$

W_{i_1} is a suitable permutation matrix. Efficient order recursive algorithms for 1-D, as well as for 2-D, MSE filtering are based on suitable partitions of the data parameters that utilize time, or spatial, shift-invariance properties [2]–[7].

The increased order linear system corresponding to the augmented support region $\mathcal{M}^{i_1}$

$$\mathcal{R}_{\mathcal{M}^{i_1}}\mathcal{C}_{\mathcal{M}^{i_1}} = -\mathcal{D}_{\mathcal{M}^{i_1}}^s$$

is partitioned using (10) as

$$\mathcal{R}_{\mathcal{M}^{i_1}} = \mathcal{W}_{i_1}^t \begin{bmatrix} r^{fo} & \hat{\mathbf{r}}_{\mathcal{M}}^{i_1 t} & r^{bo} \\ \hat{\mathbf{r}}_{\mathcal{M}}^{i_1} & \mathcal{R}_{\mathcal{M}} & \mathbf{r}_{\mathcal{M}}^{i_1} \\ r^{bo} & \mathbf{r}_{\mathcal{M}}^{i_1 t} & r^{fo} \end{bmatrix} \mathcal{W}_{i_1} \tag{11}$$

where $r^{fo} = \rho(0, 0)$, $r^{bo} = \rho(2i_1, 2k_2(i_1) + 2)$, and

$$\mathbf{r}_{\mathcal{M}}^{i_1} = \mathcal{E}[\mathcal{X}_{\mathcal{M}}(n_1, n_2)x(n_1 - i_1, n_2 - k_2(i_1) - 1)] \tag{12}$$

$$\hat{\mathbf{r}}_{\mathcal{M}}^{i_1} = \mathcal{E}[\mathcal{X}_{\mathcal{M}}(n_1, n_2)x(n_1 + i_1, n_2 + k_2(i_1) + 1)]. \tag{13}$$

Clearly, $\hat{\mathbf{r}}_{\mathcal{M}}^{i_1} = \mathcal{J}\mathbf{r}_{\mathcal{M}}^{i_1}$. Moreover,

$$\mathcal{D}_{\mathcal{M}^{i_1}}^s = \mathcal{W}_{i_1}^t \begin{bmatrix} d^s(i_1, k_2(i_1) + 1) \\ \mathcal{D}_{\mathcal{M}}^s \\ d^s(i_1, k_2(i_1) + 1) \end{bmatrix} \tag{14}$$

where $d^s(i_1, k_2(i_1) + 1) = 1/2(d(i_1, k_2(i_1) + 1) + d(-i_1, -k_2(i_1) - 1))$.

Application of the matrix inversion lemma for partitioned matrices [5] leads to a recursive estimation of the increased order filter, (1)–(5) of Table I. Auxiliary vector $\mathbf{q}_{\mathcal{M}}^{i_1}$ is defined as

$$\mathcal{R}_{\mathcal{M}}\mathbf{q}_{\mathcal{M}}^{i_1} = -\left(\mathbf{r}_{\mathcal{M}}^{i_1} + \mathcal{J}\mathbf{r}_{\mathcal{M}}^{i_1}\right) \tag{15}$$

or

$$\mathbf{q}_{\mathcal{M}}^{i_1} = \mathbf{b}_{\mathcal{M}}^{i_1} + \mathcal{J}\mathbf{b}_{\mathcal{M}}^{i_1}, \quad \mathcal{R}_{\mathcal{M}}\mathbf{b}_{\mathcal{M}}^{i_1} = -\mathbf{r}_{\mathcal{M}}^{i_1}. \tag{16}$$

The development of an order-recursive algorithm for the determination of the optimum filter $\mathcal{C}_{\mathcal{M}^{i_1}}$, for all possible neighboring directions $\{(i_1, k_2(i_1) + 1)\} \cup \{(-i_1, -k_2(i_1) - 1)\}$, $i_1 \in [-k_1, k_1]$, requires recursions for updating $\mathbf{q}_{\mathcal{M}}^{\ell}$, or equivalently, $\mathbf{b}_{\mathcal{M}}^{\ell}$, $\ell = -k_1 \cdots k_1$.

$$\mathcal{X}_{\mathcal{M}}(n_1, n_2) = \left[\mathbf{x}_{m(-k_1)}^t(n_1, n_2) \quad \mathbf{x}_{m(-k_1+1)}^t(n_1, n_2) \quad \cdots \quad \mathbf{x}_{m(k_1-1)}^t(n_1, n_2) \quad \mathbf{x}_{m(k_1)}^t(n_1, n_2)\right]^t \tag{4}$$

$$\mathcal{C}_{\mathcal{M}} = \left[\mathbf{c}_{m(-k_1)}^t \quad \mathbf{c}_{m(-k_1+1)}^t \quad \cdots \quad \mathbf{c}_{m(k_1-1)}^t \quad \mathbf{c}_{m(k_1)}^t\right]^t \tag{5}$$

$$\begin{aligned}\mathbf{x}_{m(i_1)}(n_1, n_2) &= [x(n_1 - i_1, n_2 + k_2(-i_1)) \quad \cdots \quad x(n_1 - i_1, n_2 - k_2(i_1) + 1) \quad x(n_1 - i_1, n_2 - k_2(i_1))]^t \\ \mathbf{c}_{m(i_1)} &= \left[c_{i_1, -k_2(-i_1)} \quad \cdots \quad c_{i_1, k_2(i_1)}\right]^t.\end{aligned} \tag{5a}$$

A. Auxiliary Variables Order Updating Recursions

Let us consider the increased order linear system corresponding to the augmented mask $\mathcal{M}^{i_1}$:

$$\mathcal{R}_{\mathcal{M}^{i_1}} \mathbf{b}^{\ell}_{\mathcal{M}^{i_1}} = -\mathbf{r}^{\ell}_{\mathcal{M}^{i_1}}, \qquad \ell = -k_1 \cdots k_1 \tag{17}$$

where $\mathbf{r}^{\ell}_{\mathcal{M}^{i_1}} = \mathcal{E}[\mathcal{X}_{\mathcal{M}^{i_1}}(n_1, n_2) x(n_1 - \ell, n_2 - k_2(\ell) - 1 - \delta(\ell - i_1))]$. $\delta(n)$ is the discrete-time Dirac function, i.e., $\delta(n) = 1$ for $n = 0$ and $\delta(n) = 0$ for $n \neq 0$. Clearly, $\mathbf{b}^{\ell}_{\mathcal{M}^{i_1}}$ is the backward predictor that minimizes the cost function

$$\mathcal{E}\left[\left(x(n_1 - \ell, n_2 - k_2(\ell) - 1 - \delta(\ell - i_1)) + \mathcal{X}^{t}_{\mathcal{M}}(n_1, n_2)\mathbf{b}^{i_1}_{\mathcal{M}}\right)^2\right].$$

In contrast to the order updating recursion derived for the optimum filter (9), the update of the backward predictors defined above requires a more complicated procedure. The main difference between (9) and (16) is that, while in the first case the right-hand side vector is symmetric, in the second, it is not. This time, a two-step order-updating method will be followed, as

$$\begin{aligned} \mathbf{b}^{\ell}_{\mathcal{M}} &\to \mathbf{b}^{\ell}_{\mathcal{M}+L(i_1)}, \qquad \forall \ell \in [-k_1, k_1] \quad \text{step I} \\ \mathbf{b}^{\ell}_{\mathcal{M}+L(i_1)} &\to \mathbf{b}^{\ell}_{\mathcal{M}^{i_1}}, \qquad \forall \ell \in [-k_1, k_1] \quad \text{step II} \end{aligned}$$

Steps I and II described above correspond to a two-step incremental update of the initial mask $\mathcal{M}$ as

$$\begin{aligned} \mathcal{M} &\to (\mathcal{M} \cup (i_1, k_2(i_1) + 1)) \equiv \mathcal{M} + L(i_1) \\ \mathcal{M} + L(i_1) &\to (\mathcal{M} + L(i_1) \cup \{(-i_1, -k_2(i_1) - 1)) \equiv \mathcal{M}^{i1}. \end{aligned}$$

Let us first consider step II. The increased order data vector is partitioned as

$$\mathcal{X}_{\mathcal{M}^{i_1}}(n_1, n_2) = \mathcal{T}^{t}_{R(-i_1)} \begin{bmatrix} x(n_1 + i_1, n_2 + k_2(i_1) + 1) \\ \mathcal{X}_{\mathcal{M}+L(i_1)}(n_1, n_2) \end{bmatrix}. \tag{18}$$

$\mathcal{T}^{t}_{R(-i_1)}$ is a suitable permutation matrix. Thus, parameters appearing in (16) are partitioned as

$$\begin{aligned} \mathcal{R}_{\mathcal{M}^{i_1}} &= \mathcal{T}^{t}_{R(-i_1)} \begin{bmatrix} \rho(0,0) & \hat{\mathbf{r}}^{i_1 t}_{\mathcal{M}+L(i_1)} \\ \hat{\mathbf{r}}^{i_1}_{\mathcal{M}+L(i_1)} & \mathcal{R}_{\mathcal{M}+L(i_1)} \end{bmatrix} \mathcal{T}_{R(-i_1)} \\ \mathbf{r}^{\ell}_{\mathcal{M}^{i_1}} &= \mathcal{T}^{t}_{R(-i_1)} \begin{bmatrix} \rho(i_1 + \ell, k_2(i_1) + k_2(\ell) + 2 + \delta(\ell - i_1)) \\ \mathbf{r}^{\ell}_{\mathcal{M}+L(i_1)} \end{bmatrix} \end{aligned}$$

where

$$\begin{aligned} \hat{\mathbf{r}}^{i_1}_{\mathcal{M}+L(i_1)} &= \mathcal{E}[\mathcal{X}_{\mathcal{M}+L(i_1)}(n_1, n_2) x(n_1 + i_1, n_2 + k_2(i_1) + 1] \\ \mathbf{r}^{\ell}_{\mathcal{M}+L(i_1)} &= \mathcal{E}[\mathcal{X}_{\mathcal{M}+L(i_1)}(n_1, n_2) x(n_1 - \ell, n_2 - k_2(\ell) - 1 \\ &\qquad - \delta(\ell - i_1))]. \end{aligned}$$

Application of the matrix inversion lemma results in (19)–(21) of Table I. The new parameter introduced, $\mathbf{a}^{i_1}_{\mathcal{M}+L(i_1)}$, is the forward predictor, defined as

$$\mathcal{R}_{\mathcal{M}+L(i_1)} \mathbf{a}^{i_1}_{\mathcal{M}+L(i_1)} = -\hat{\mathbf{r}}^{i_1}_{\mathcal{M}+L(i_1)}.$$

It can be interpreted as a forward predictor that minimizes the cost function

$$\begin{aligned} \mathcal{E}\Big[\Big(&x(n_1 + i_1, n_2 + k_2(i_1) + 1) \\ &+ \mathcal{X}^{t}_{\mathcal{M}+L(i_1)}(n_1, n_2)\mathbf{a}^{i_1}_{\mathcal{M}+L(i_1)}\Big)^2\Big]. \end{aligned}$$

It is updated according to the recursion described by (15)–(17) of Table I. Due to the perisymmetric property of matrix $\mathcal{R}_{\mathcal{M}}$ and the symmetry of (12), (13), the lower order forward predictors that correspond to the support region $\mathcal{M}$ are symmetric to the backward predictors, i.e.,

$$\mathbf{a}^{\ell}_{\mathcal{M}} = \mathbf{J}\mathbf{b}^{\ell}_{\mathcal{M}}, \qquad \ell \in [-k_1, k_1].$$

The implementation of step I needs more care. Suppose that $\ell \neq i_1$; the case where $\ell = i_1$ is treated separately. Indeed, following (18), we obtain

$$\mathcal{X}_{\mathcal{M}+L(i_1)}(n_1, n_2) = \mathcal{S}^{t}_{L(i_1)} \begin{bmatrix} \mathcal{X}_{\mathcal{M}}(n_1, n_2) \\ x(n_1 - i_1, n_2 - k_2(i_1) - 1) \end{bmatrix}. \tag{19}$$

$\mathcal{S}_{L(i_1)}$ is a permutation matrix. Moreover,

$$\mathcal{R}_{\mathcal{M}+L(i_1)} = \mathcal{S}^{t}_{L(i_1)} \begin{bmatrix} \mathcal{R}_{\mathcal{M}} & \mathbf{r}^{i_1}_{\mathcal{M}+L(i_1)} \\ \mathbf{r}^{i_1 t}_{\mathcal{M}+L(i_1)} & \rho(0,0) \end{bmatrix} \mathcal{S}_{L(i_1)}$$

and

$$\hat{\mathbf{r}}^{i_1}_{\mathcal{M}+L(i_1)} = \mathcal{S}^{t}_{L(i_1)} \begin{bmatrix} \mathcal{J}\mathbf{r}^{i_1}_{\mathcal{M}} \\ \rho(2i_1, 2k_2(i_1) + 2) \end{bmatrix}$$

and for $\ell \neq i_1$,

$$\mathbf{r}^{\ell}_{\mathcal{M}+L(i_1)} = \mathcal{S}^{t}_{L(i_1)} \begin{bmatrix} \mathbf{r}^{\ell}_{\mathcal{M}} \\ \rho(i_1 - \ell, k_2(i_1) - k_2(\ell)) \end{bmatrix}.$$

Thus, (7)–(8) of Table I are easily derived using the matrix inversion lemma.

When $\ell = i_1$, partition (19) does not hold. To overcome this problem, the data vector $\mathcal{X}_{\mathcal{M}+L(i_1)}(n_1, n_2)$ must, somehow, be related to the delayed version of the lower order data vector $\mathcal{X}_{\mathcal{M}}(n_1, n_2 - 1)$. To accomplish this task, the following augmented mask is considered:

$$\mathcal{M} + \mathbf{L} = \mathcal{M} \bigcup_{i_1 = -k_1}^{k_1} \{(i_1, k_2(i_1) + 1)\}. \tag{20}$$

The corresponding data vector is partitioned in two ways as

$$\begin{aligned} \mathcal{X}_{\mathcal{M}+L}(n_1, n_2) &= \mathcal{T}^{t}_{L} \begin{bmatrix} \mathbf{x}^{f}(n_1, n_2 - 1) \\ \mathcal{X}_{\mathcal{M}}(n_1, n_2 - 1) \end{bmatrix} \\ &= \mathcal{S}^{t}_{L} \begin{bmatrix} \mathcal{X}_{\mathcal{M}+L(i_1)}(n_1, n_2) \\ \mathbf{x}^{b}_{i_1}(n_1, n_2) \end{bmatrix}. \end{aligned} \tag{21}$$

Vectors $\mathbf{x}^{f}(n_1, n_2)$ and $\mathbf{x}^{b}_{i_1}(n_1, n_2)$ are defined by

$$\begin{aligned} \mathbf{x}^{f}(n_1, n_2) &= [x(n_1 + \ell, n_2 + k_2(-\ell))]_{\ell = -k_1 \cdots k_1}, \\ \mathbf{x}^{b}_{i_1}(n_1, n_2) &= [x(n_1 - \ell, n_2 - k_2(\ell) - 1)]_{\substack{\ell = -k_1 \cdots k_1 \\ \ell \neq i_1}}. \end{aligned}$$

$\mathcal{T}_L$ and $\mathcal{S}_L$ are suitable permutation matrices.

Thus,

$$\begin{aligned} \mathcal{R}_{\mathcal{M}+L} &= \mathcal{T}^{t}_{L} \begin{bmatrix} R^{fo} & \hat{\mathbf{R}}^{t}_{\mathcal{M}} \\ \hat{\mathbf{R}}_{\mathcal{M}} & \mathcal{R}_{\mathcal{M}} \end{bmatrix} \mathcal{T}_L = \mathcal{S}^{t}_{L} \begin{bmatrix} \mathcal{R}_{\mathcal{M}+L(i_1)} & \mathbf{R}^{i_1}_{\mathcal{M}} \\ \mathbf{R}^{i_1 t}_{\mathcal{M}} & R^{bo}_{i_1} \end{bmatrix} \mathcal{S}_L \\ \mathbf{r}^{i_1}_{\mathcal{M}+L} &= \mathcal{T}^{t}_{L} \begin{bmatrix} \rho^{f} \\ \mathbf{r}^{i_1}_{\mathcal{M}} \end{bmatrix} = \mathcal{S}^{t}_{L} \begin{bmatrix} \mathbf{r}^{i_1}_{\mathcal{M}+L(i_1)} \\ \rho^{b(i_1)} \end{bmatrix} \end{aligned}$$

where

$$\begin{aligned} R^{fo} &= [\rho(\ell_1 - \ell_1, k_2(-\ell_1) - k_2(-\ell_2))]_{\substack{\ell_1 = -k_1 \cdots k_1 \\ \ell_2 = -k_1 \cdots k_1}} \\ \hat{\mathbf{R}}_{\mathcal{M}} &= \left[\mathcal{J}\mathbf{r}^{\ell}_{\mathcal{M}}\right]_{\ell = -k_1 \cdots k_1}, \\ \mathbf{R}_{\mathcal{M}+L(i_1)} &= \left[\mathbf{r}^{\ell}_{\mathcal{M}+L(i_1)}\right]_{\substack{\ell = -k_1 \cdots k_1 \\ \ell \neq i_1}} \\ \rho^{b(i_1)} &= [\rho(\ell - i_1, k_2(\ell) - k_2(i_1) - 2)]_{\substack{\ell = -k_1 \cdots k_1 \\ \ell \neq i_1}}, \\ \rho^{f} &= [\rho(\ell - i_1, -k_2(-\ell) - k_2(i_1) - 1)]_{\substack{\ell = -k_1 \cdots k_1 \\ \ell \neq i_1}}. \end{aligned}$$

The above partitions yield (10)–(17) of Table I.

TABLE I
THE ALGORITHM

$$\mathbf{q}^{i_1}_{\mathcal{M}} = \mathbf{b}^{i_1}_{\mathcal{M}} + \mathcal{J}\mathbf{b}^{i_1}_{\mathcal{M}} \tag{1}$$

$$\beta_{i_1} = 1/2(d(i_1, k_2(i_1)+1) + d(-i_1, -k_2(i_1)-1)) + \mathbf{r}^{i_1 t}_{\mathcal{M}}\mathcal{C}_{\mathcal{M}} \tag{2}$$

$$\alpha^{s}_{i_1} = \rho(0,0) + \rho(2i_1, 2k_2(i_1)+2) + \mathbf{r}^{i_1 t}_{\mathcal{M}}\mathbf{q}^{i_1}_{\mathcal{M}} \tag{3}$$

$$k_{i_1} = -\beta_{i_1}/\alpha^{s}_{i_1} \tag{4}$$

$$\mathcal{W}_{i_1}\mathcal{C}_{\mathcal{M}^{i_1}} = \begin{bmatrix} 0 \\ \mathcal{C}_{\mathcal{M}} \\ 0 \end{bmatrix} + \begin{bmatrix} 1 \\ \mathbf{q}^{i_1}_{\mathcal{M}} \\ 1 \end{bmatrix} k_{i_1} \tag{5}$$

$$\alpha_{i_1} = \rho(0,0) + \mathbf{r}^{i_1}_{\mathcal{M}}\mathbf{b}^{i_1}_{\mathcal{M}} \tag{6}$$

```
FOR ℓ = -k1 TO k1, AND ℓ ≠ i1, DO
```

$$\beta^{b\ell}_{i_1} = \rho(i_1-\ell, k_2(i_1)-k_2(\ell)) + \mathbf{r}^{i_1}_{\mathcal{M}}\mathbf{b}^{\ell}_{\mathcal{M}} \tag{7}$$

$$k^{b\ell}_{i_1} = -\beta^{b\ell}_{i_1}/\alpha_{i_1} \tag{8}$$

$$\mathcal{S}_{i_1}\mathbf{b}^{\ell}_{\mathcal{M}+L(i_1)} = \begin{bmatrix} \mathbf{b}^{\ell}_{\mathcal{M}} \\ 0 \end{bmatrix} + \begin{bmatrix} \mathbf{b}^{i_1}_{\mathcal{M}} \\ 1 \end{bmatrix} k^{b\ell}_{i_1} \tag{9}$$

```
ENDFOR ℓ
```

$$\boldsymbol{\mathcal{A}} = \left[\mathcal{J}\mathbf{b}^{\ell}_{\mathcal{M}}\right]_{\ell=-k_1\ldots k_1}, \boldsymbol{\mathcal{B}}^{i_1} = \left[\mathbf{b}^{\ell}_{\mathcal{M}+L(i_1)}\right]_{\substack{\ell=-k_1\ldots k_1 \\ \ell\neq i_1}}$$

$$\beta^{b(i_1)}_{L} = \boldsymbol{\rho}^{b(i_1)}_{L} + \boldsymbol{\mathcal{A}}^{t}_{\mathcal{M}}\mathbf{r}^{i_1}_{\mathcal{M}} \tag{10}$$

$$\boldsymbol{\alpha}_L = R^{fo} + \widehat{\mathbf{R}}^{t}_{\mathcal{M}}\boldsymbol{\mathcal{A}}_{\mathcal{M}} \tag{11}$$

$$K^{b(i_1)}_{L} = -\boldsymbol{\alpha}^{-1}_{L}\beta^{b(i_1)}_{L} \tag{12}$$

$$\mathcal{T}_L\mathbf{b}^{i_1}_{\mathcal{M}+L} = \begin{pmatrix} \mathbf{0} \\ \mathbf{b}^{i_1}_{\mathcal{M}} \end{pmatrix} + \begin{pmatrix} \mathbf{I} \\ \boldsymbol{\mathcal{A}}_{\mathcal{M}} \end{pmatrix} K^{b(i_1)}_{L} \tag{13}$$

$$\begin{pmatrix} \mathbf{b}^{i_1}_{\mathcal{M}+L(i_1)} \\ \mathbf{0} \end{pmatrix} = \boldsymbol{\mathcal{S}}_R\mathbf{b}^{i_1}_{\mathcal{M}+L} - \begin{pmatrix} \boldsymbol{\mathcal{B}}^{i_1}_{\mathcal{M}+L(i_1)} \\ \mathbf{I} \end{pmatrix} \widehat{K}^{b(i_1)}_{L} \tag{14}$$

$$\beta^{f}_{i_1} = \rho(2i_1, 2k_2(i_1)+2) + \mathbf{r}^{i_1}_{\mathcal{M}}\mathcal{J}\mathbf{b}^{i_1}_{\mathcal{M}} \tag{15}$$

$$k^{f}_{i_1} = -\beta^{f}_{i_1}/\alpha_{i_1} \tag{16}$$

$$\mathcal{S}_{i_1}\mathbf{a}^{i_1}_{\mathcal{M}+L(i_1)} = \begin{bmatrix} \mathcal{J}\mathbf{b}^{i_1}_{\mathcal{M}} \\ 0 \end{bmatrix} + \begin{bmatrix} \mathbf{b}^{i_1}_{\mathcal{M}} \\ 1 \end{bmatrix} k^{f}_{i_1} \tag{17}$$

$$\alpha_{-i_1} = \rho(0,0) + \widehat{\mathbf{r}}^{\ell}_{\mathcal{M}+L(i_1)}\mathbf{a}^{i_1}_{\mathcal{M}+L(i_1)} \tag{18}$$

```
FOR ℓ = -k1 TO k1, DO
```

$$\beta^{b\ell}_{-i_1} = \rho(i_1+\ell, k_2(i_1)+k_2(\ell)+2+\delta(\ell-i_1)) + \widehat{\mathbf{r}}^{i_1}_{\mathcal{M}+L(i_1)}\mathbf{b}^{i_1}_{\mathcal{M}+L(i_1)} \tag{19}$$

$$k^{b\ell}_{-i_1} = \beta^{b\ell}_{-i_1}/\alpha_{-i_1} \tag{20}$$

$$\mathcal{T}_{R(-i)}\mathbf{b}^{\ell}_{\mathcal{M}^{i_1}} = \begin{bmatrix} 0 \\ \mathbf{b}^{\ell}_{\mathcal{M}+L(i_1)} \end{bmatrix} + \begin{bmatrix} 1 \\ \mathbf{a}^{i_1}_{\mathcal{M}+L(i_1)} \end{bmatrix} k^{b\ell}_{-i_1} \tag{21}$$

```
ENDFOR ℓ
```

B. Overall Organization

The order-recursive equations developed so far are tied together to form a powerful *true order-recursive* 2-D algorithm for symmetric filtering and linear prediction. Indeed, let $\mathcal{M}^{\text{fin}}$ be the support region within which the search for the optimum mask will be conducted. Let $k_1^{\text{fin}} = \max\{i_1 : (i_1, i_2) \in \mathcal{M}^{\text{fin}}\}$. Then, for all $i_1 \in [-k_1, k_1], k_1 \leq k_1^{\text{fin}}$, any one of the increased order filters corresponding to a symmetric increment along the i_1 and the $-i_1$ rows of $\mathcal{M}$ can be estimated as $\mathbf{C}_{\mathcal{M}} \rightarrow \mathbf{C}_{\mathcal{M}^{i_1}}$. The update of parameters to a mask that contains extra rows, i.e., going from $[-k_1, k_1] \rightarrow [-k_1-1, k_1+1]$, can be accomplished only for the points laying across the vertical axis. Once the increased order filter that corresponds to $\mathcal{M} \cup \{(0, -k_1-1)\} \cup \{(0, k_1+1)\}$ is determined, further recursions along that row can be performed.

The computational complexity of the algorithm is $O(2k_1P)$ operations per recursion, where $P = \dim(\mathcal{C}_{\mathcal{M}}) = \sum_{i_1=0}^{k_1} 2k_2(i_1)+1$. For a 2-D filter of a final mask shape $\mathcal{M}^{\text{fin}}$, $O(k_1^{\text{fin}}(P^{\text{fin}})^2)$ operations

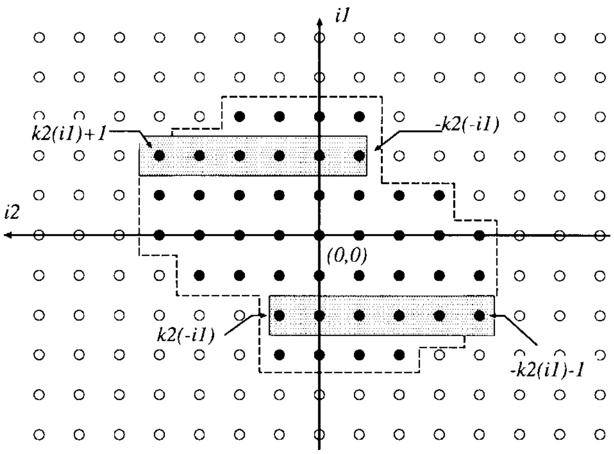

Fig. 2. Extended support region corresponding to an increased order symmetric filter.

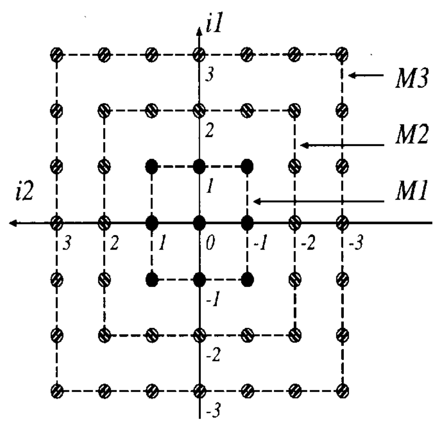

Fig. 3. Example 1.

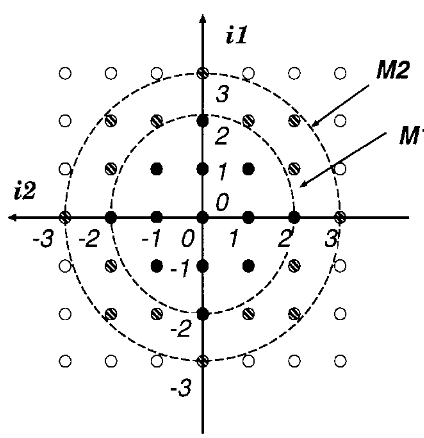

Fig. 4. Example 2.

are required. For the special case of rectangular-shaped masks, this amount is comparable to the complexity of the algorithms proposed in [2]–[4]. The nonrectangular mask case cannot be handled by the algorithms of [2]–[4] unless overparametrization is utilized.

A great advantage the proposed algorithm offers against conventional LWR based counterparts is the accommodation of masks of general boundaries and the estimation of lower order parameters. Moreover, all lower order filters that correspond to reduced shape masks can be recovered. Consider, for example, a filter mask of a rectangular shape, $M = (-k^{\text{fin}}, -k^{\text{fin}}) \times (k^{\text{fin}}, k^{\text{fin}})$. When all filters of intermediate order $(-k, -k) \times (k, k)$ are required, for all $1 \leq k \leq k^{\text{fin}}$, algorithms of [2]–[4] require a repetitive application of LWR-based recursions, thus resulting in $O(k^6)$ cost, which is outperformed by the $O(k^5)$ cost of the proposed method. Thus, it is established that in all cases, the proposed highly efficient order-recursive 2-D algorithm performs better than any existing scheme.

The order-updating procedure is illustrated using a simple but important support region depicted in Fig. 3. Suppose that the MSE filter corresponding to $\mathcal{M}_1$ is known. The estimation of the increased order MSE filter corresponding to $\mathcal{M}_2$ is accomplished following the updating scheme

$$\begin{aligned} M1 &\cup \{(2,0),(-2,0)\} \\ &\cup \{(2,1),(-2,-1)\} \\ &\cup \{(2,2),(-2,-2)\} \\ &\cup \{(1,2),(-1,-2)\} \\ &\cup \{(0,2),(0,-2)\} \\ &\cup \{(-1,2),(1,-2)\} \\ &\cup \{(-2,2),(2,-2)\} \\ &\cup \{(-2,1),(2,-1)\}. \end{aligned}$$

A case of a circularly shaped filter mask is depicted in Fig. 4. The optimum filter corresponding to the mask $\mathcal{M}_2$ is estimated from the lower order counterpart as

$$\begin{aligned} M1 &\cup \{(3,0),(-3,0)\} \\ &\cup \{(2,1),(-2,-1)\} \\ &\cup \{(1,2),(-1,-2)\} \\ &\cup \{(2,2),(-2,-2)\} \\ &\cup \{(1,2),(-1,-2)\} \\ &\cup \{(0,3),(0,-3)\} \\ &\cup \{(-1,2),(1,-2)\} \\ &\cup \{(-2,1),(2,-2)\} \\ &\cup \{(-2,2),(2,-2)\}. \end{aligned}$$

IV. Conclusion

A highly efficient, order-recursive algorithm for symmetric 2-D FIR filtering and 2-D system identification has been developed. Symmetric support regions with arbitrary horizontally convex shape can be handled. The proposed algorithm offers the greatest possible maneuverability in the 2-D index space. It allows for recursive estimation of the 2-D filter mask shape. The implicit flexibility of the algorithm enables a dynamical reconfiguration of the mask shape in a computationally efficient way.

References

[1] J. Lim, *Two-Dimensional Signal and Image Processing.* Englewood Cliffs, NJ: Prentice-Hall, 1990.

[2] A. Kok, D. Manolakis, and V. Ingle, "Symmetric noncausal spatial models for 2-D signals with applications in stochastic texture modeling," *Multidimensional Syst. Signal Processing,* pp. 125–149, 1993.

[3] N. Kalouptsidis, "Efficient computation of multichannel Wiener filters with linear phase," *IEEE Trans. Circuits Syst.,* vol. 35, pp. 433–438, Apr. 1988.

[4] D. Manolakis and V. Ingle, "Fast algorithms for 2-D FIR Wiener filtering and linear prediction," *IEEE Trans. Circuits Syst.,* vol. CAS-34, pp. 181–185, Feb. 1987.

[5] N. Kalouptsidis and S. Theodoridis, *Adapive System Identification and Signal Processing Algorithms.* Englewood Cliffs, NJ: Prentice-Hall, 1993.

[6] G. Glentis and N. Kalouptsidis, "Efficient order recursive algorithms for multichannel least squares filtering," *IEEE Trans. Signal Processing*, vol. 40, pp. 1354–1374, June 1992.
[7] ——, "Efficient multichannel FIR filtering using a single step versatile order recursive algorithm," *Signal Processing*, vol. 37, pp. 437–462, 1994.
[8] R. Kashyap and R. Chellappa, "Estimation and choice of neighbors in spatial-interaction models of images," *IEEE Trans. Inform. Theory*, vol. IT-29, pp. 60–72, Jan. 1983.
[9] P. Djuric and S. Kay, "Model order estimation of 2D autoregressive processes," in *Proc. IEEE ICASSP*, 1991, pp. 3405–3408.

Efficient Moving-Window DFT Algorithms

J. A. Rosendo Macías and A. Gómez Expósito

***Abstract*—This brief deals with the real-valued, moving-window discrete Fourier transform. After reviewing the basic recursive versions appearing in the literature, additional recursive equations are presented. Then, these equations are combined so that nonrecursive expressions involving only consecutive discrete Fourier transform (DFT) sine components are obtained for both the DFT cosine component and squared harmonic amplitude.**

The computational complexity of this new scheme is finally studied and compared to that of existing methods, showing that, in most practical situations, a reduction in the operation count is achieved.

***Index Terms*—Discrete Fourier transform, harmonic analysis, recursive digital filters, spectral analysis.**

I. Introduction

Discrete Fourier-related transforms are of paramount importance in many and diverse scientific and technological applications, constituting one of the major branches embraced in digital signal processing techniques. A recent and thorough discussion of major continuous-time and discrete-time Fourier-related transforms and series, both for real and complex signals, is presented in [1].

Since the paper by Cooley and Tukey introducing the fast Fourier transform (FFT) concept was published in 1965 [2], a formidable research effort has been devoted to the development of increasingly efficient fast transform algorithms [3]. Techniques such as radix-p ($p = 2, 4, 8$), split-radix, DFT via other transforms, mixed decimation, etc., have been proposed within this framework.

Although earlier FFT algorithms were designed to deal with complex sequences, it turns out that, in many important applications, the sequence to be transformed is real valued. Consequently, several fast transform techniques specifically devised for real signals have been proposed [1], [4]. Among these, the real split-radix FFT [4], the so-called fast Fourier-cosine transform, based on a recursive application of the DCT [5], and the fast Fourier-sine transform, based on the same idea but using the DST [6], have been shown to require the lowest operation count.

An interesting particular case appears when the spectrum of a time-varying or nonstationary signal is required. This leads to the so-called short-time, time-dependent or moving-window Fourier transform [1], [7]. As its name implies, the DFT is computed on a window of the signal which is moved successively one data point each time. This way, there is a simple relationship between the window contents before and after the shift since, as the oldest sample leaves the window, a single new point is added to it. Although a straightforward rectangular window is implicit in this process, other more sophisticated windows have been proposed in digital filtering to avoid unacceptable edge effects [8].

In spite of the practical importance of this type of DFT, the effort devoted to design algorithms especially tailored to the changing window contents is rather modest. Virtually all proposals in this regard refer to recursive algorithms, in which the spectrum at any time instant is simply related to previously computed spectra [9]–[11]. However, since practical recursive implementations are prone to finite-length, long-term anomalous behavior [8, p. 308], [12], [13], the need for efficient, nonrecursive, moving-window DFT algorithms still remains.

In this paper, the real-valued, moving-window DFT is first reviewed. Then, first- and second-order recursive versions are obtained in a simple manner [10], [13]. These recursive versions are subsequently used to develop nonrecursive versions of the DFT in which only a single component at consecutive instants is required to fully compute the complex spectrum. Finally, the computational effort of these improved, reliable versions is compared to that of the best, general-purpose, real-valued FFT algorithms.

II. Moving-Window DFT

Assuming that, at instant n, the window comprises the N values $x(n-N+1), x(n-N+2), \cdots, x(n)$, the kth complex harmonic, as given by the DFT, is

$$\mathcal{F}_n(k) = \sum_{i=0}^{N-1} x(i+n-N+1)e^{-j\theta_k i} \tag{1}$$

where

$$\theta_k = \frac{2\pi k}{N}. \tag{2}$$

Using rectangular coordinates, and letting $S_n(k)$, $C_n(k)$ be the DFT sine and cosine, components, respectively, i.e.,

$$C_n(k) = \sum_{i=0}^{N-1} x(i+n-N+1)\cos(i\theta_k) \tag{3}$$

$$S_n(k) = -\sum_{i=0}^{N-1} x(i+n-N+1)\sin(i\theta_k) \tag{4}$$

the DFT harmonics can be expressed as

$$\mathcal{F}_n(k) = C_n(k) + jS_n(k) \tag{5}$$

and their amplitudes or power spectrum

$$F_n^2(k) = C_n^2(k) + S_n^2(k). \tag{6}$$

Taking advantage of the shifting window contents, a recursive expression for the DFT can be obtained with $O(N)$ complexity [9], [11]:

$$\mathcal{F}_n(k) = [\mathcal{F}_{n-1}(k) + \Delta x(n)]e^{j\theta_k} \tag{7}$$

where $\Delta x(n) = x(n) - x(n-N)$.

Manuscript received September 27, 1996; revised March 18, 1997. This paper was recommended by Associate Editor N. K. Bose.

The authors are with the Department of Electrical Engineering, University of Seville, 41012 Seville, Spain (e-mail: exposito@cica.es).

Publisher Item Identifier S 1057-7130(98)01641-3.