terms involving one variable in an MV polynomial are fixed, the coefficients of the remaining terms in the polynomial are rigidly related to these, if the polynomial has to be separable.

ACKNOWLEDGMENT

The author wishes to thank Dr. K. S. Shanmugam and S. Rai for valuable assistance.

REFERENCES

[1] J. D. Rhodes, "Two-variable transmission line networks," in *Proc. 1974 IEEE Int. Symp. Circuits and Systems*, pp. 95-99, Apr. 1974.
[2] J. L. Shanks, S. Treitel, and J. H. Justice, "Stability and synthesis of 2-D recursive filters," *IEEE Trans. Audio and Electroacoust.*, vol. AU-20, pp. 115-128, June 1972.
[3] S. Treitel and J. L. Shanks, "The design of multistage separable planar filters," *IEEE Trans. Geosci. Electron.*, vol. GE-9, pp. 10-27, Jan. 1971.
[4] J. J. Bussgang, L. Ehrman, and J. W. Graham, "Analysis of nonlinear systems with multiple inputs," *Proc. IEEE*, vol. 62, pp. 1088-1119, Aug. 1974.

The Complex LMS Algorithm

BERNARD WIDROW, JOHN McCOOL, AND MICHAEL BALL

***Abstract*—A least-mean-square (LMS) adaptive algorithm for complex signals is derived. The original Widrow-Hoff LMS algorithm is $W_{j+1} = W_j + 2\mu\epsilon_j X_j$. The complex form is shown to be $\mathbf{W}_{j+1} = \mathbf{W}_j + 2\mu\boldsymbol{\epsilon}_j \bar{\mathbf{X}}_j$, where the boldfaced terms represent complex (phasor) signals and the bar above $\bar{\mathbf{X}}_j$ designates complex conjugate.**

The adaptive linear combiner is the key element in many adaptive systems. Its function is to weight and sum a set of input signals to form an adaptive output. The input signal vector X and the weight vector W are defined at time j as follows:

$$X_j = \begin{Bmatrix} x_{1j} \\ x_{2j} \\ \cdot \\ \cdot \\ \cdot \\ x_{nj} \end{Bmatrix} \qquad W_j = \begin{Bmatrix} w_{1j} \\ w_{2j} \\ \cdot \\ \cdot \\ \cdot \\ w_{nj} \end{Bmatrix}. \tag{1}$$

The input signals are sampled (i.e., discrete in time), and the weights are alterable. The output at time j is

$$y_j = X_j^T W_j = W_j^T X_j. \tag{2}$$

The error signal ϵ_j required for adaptation is defined as the difference between the desired response d_j (an externally supplied input) and the output y_j:

$$\epsilon_j = d_j - y_j = d_j - W_j^T X_j. \tag{3}$$

The least-mean-square (LMS) adaptive algorithm [1]-[3] minimizes the mean-square error ϵ_j by recursively altering the weight vector W_j at each sampling instant according to the expression

$$W_{j+1} = W_j + 2\mu\epsilon_j X_j \tag{4}$$

where μ is a convergence factor controlling stability and rate of adaptation. The algorithm is based on the method of steepest descent, moving W_j in proportion to the instantaneous gradient estimate of the mean square error. A number of convergence proofs, derivations of performance characteristics, and applications have appeared in [4]-[7].

Some applications of the adaptive linear combiner require a complex output. These include the adaptive filtering of high-frequency narrowband signals at an intermediate frequency, in which case both X_j and d_j are translated in frequency without changing their phase relationships.

Fig. 1 shows two ways of representing a complex adaptive linear combiner. The complex input vector $\mathbf{X}_j$ and complex weight vector $\mathbf{W}_j$ are given by

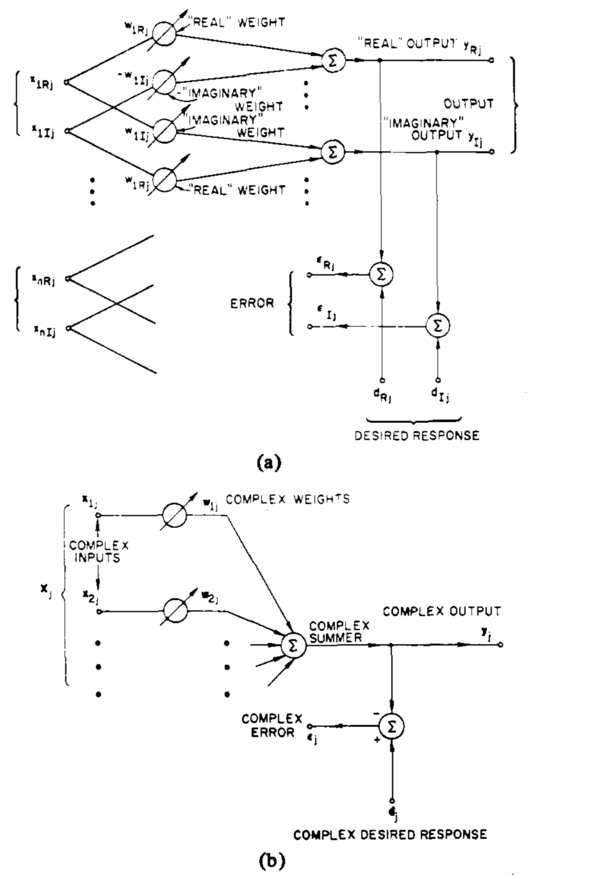

Fig. 1. Complex adaptive linear combiner. (a) In block diagram form. (b) In schematic representation.

$$\mathbf{X}_j \triangleq \begin{Bmatrix} x_{1Rj} \\ x_{2Rj} \\ \cdot \\ \cdot \\ \cdot \\ x_{nRj} \end{Bmatrix} + i \begin{Bmatrix} x_{1Ij} \\ x_{2Ij} \\ \cdot \\ \cdot \\ \cdot \\ x_{nIj} \end{Bmatrix} = X_{Rj} + iX_{Ij}$$

$$\mathbf{W}_j \triangleq \begin{Bmatrix} w_{1Rj} \\ w_{2Rj} \\ \cdot \\ \cdot \\ \cdot \\ w_{nRj} \end{Bmatrix} + i \begin{Bmatrix} w_{1Ij} \\ w_{2Ij} \\ \cdot \\ \cdot \\ \cdot \\ w_{nIj} \end{Bmatrix} = W_{Rj} + iW_{Ij} \tag{5}$$

where R designates a direct (real) signal component and I a 90°-shifted (imaginary) signal component. Although it appears in Fig. 1(a) that four weights are associated with each input pair, only 2° of freedom are actually represented. The complex error and desired response required to adapt both the real and imaginary weights are given by

$$\boldsymbol{\epsilon}_j \triangleq \epsilon_{Rj} + i\epsilon_{Ij}$$
$$\mathbf{d}_j \triangleq d_{Rj} + id_{Ij}. \tag{6}$$

The complex output is correspondingly given by

$$\mathbf{y}_j \triangleq y_{Rj} + iy_{Ij}. \tag{7}$$

Equations (2) and (3) may thus be expressed in complex form as follows:

$$\mathbf{y}_j = \mathbf{X}_j^T \mathbf{W}_j = \mathbf{W}_j^T \mathbf{X}_j \tag{8}$$

$$\boldsymbol{\epsilon}_j = \mathbf{d}_j - \mathbf{y}_j = \mathbf{d}_j - \mathbf{W}_j^T \mathbf{X}_j = \mathbf{d}_j - \mathbf{X}_j^T \mathbf{W}_j. \tag{9}$$

Although these equations are more general than (2) and (3), they correspond exactly. All multiplies and adds are complex.

Manuscript received August 19, 1974; revised October 18, 1974.
B. Widrow is with the Information Systems Laboratory, Stanford University, Stanford, Calif. 94305.
J. McCool and M. Ball are with the Fleet Engineering Department, Naval Undersea Center, San Diego, Calif. 92132.

The complex LMS algorithm must be able to adapt the real and imaginary parts of W_j simultaneously, minimizing in some sense both ϵ_{Rj} and ϵ_{Ij}. A reasonable objective is to minimize the average total error power,

$$E[\epsilon_j\bar{\epsilon}_j] = E[\epsilon_{Rj}^2 + \epsilon_{Ij}^2] = E[\epsilon_{Rj}^2] + E[\epsilon_{Ij}^2] \tag{10}$$

where E designates expected value and the bar above $\bar{\epsilon}_j$ complex conjugate. Since the two components of the error are in quadrature relative to each other, they cannot be minimized independently.

The derivation of the complex LMS algorithm for minimizing $E[\epsilon_j\bar{\epsilon}_j]$ is similar to the derivation of the original LMS algorithm, except that the rules of complex algebra must be observed. The conjugate of the complex error (9) is

$$\bar{\epsilon}_j = \bar{d}_j - \bar{W}_j^T X_j = \bar{d}_j - X_j^T \bar{W}_j. \tag{11}$$

The instantaneous gradient of $\epsilon_j\bar{\epsilon}_j$ with respect to the real component of the weight vector is

$$\nabla_R(\epsilon_j\bar{\epsilon}_j) \triangleq \begin{Bmatrix} \dfrac{\partial(\epsilon_j\bar{\epsilon}_j)}{\partial w_{IR}} \\ \cdot \\ \cdot \\ \cdot \\ \dfrac{\partial(\epsilon_j\bar{\epsilon}_j)}{\partial w_{nR}} \end{Bmatrix} = \epsilon_j\nabla_R(\bar{\epsilon}_j) + \bar{\epsilon}_j\nabla_R(\epsilon_j) = \epsilon_j(-\bar{X}_j) + \bar{\epsilon}_j(-X_j). \tag{12}$$

The instantaneous gradient with respect to the imaginary component is

$$\nabla_I(\epsilon_j\bar{\epsilon}_j) = \epsilon_j\nabla_I(\bar{\epsilon}_j) + \bar{\epsilon}_j\nabla_I(\epsilon_j) = \epsilon_j(i\bar{X}_j) + \bar{\epsilon}_j(-iX_j). \tag{13}$$

Applying the method of steepest descent to the real and imaginary parts of the weight vector by changing them along their respective negative gradient estimates, one obtains

$$\begin{aligned} W_{Rj+1} &= W_{Rj} - \mu\nabla_R(\epsilon_j\bar{\epsilon}_j) \\ W_{Ij+1} &= W_{Ij} - \mu\nabla_I(\epsilon_j\bar{\epsilon}_j). \end{aligned} \tag{14}$$

Since the complex weight vector is $W_j = W_{Rj} + iW_{Ij}$, the complex weight iteration rule can be expressed as

$$W_{j+1} = W_j - \mu[\nabla_R(\epsilon_j\bar{\epsilon}_j) + i\nabla_I(\epsilon_n\bar{\epsilon}_n)]. \tag{15}$$

If the gradients (12) and (13) are now substituted in (15), the complex form of the LMS algorithm results:

$$W_{j+1} = W_j + 2\mu\epsilon_j\bar{X}_j. \tag{16}$$

REFERENCES

[1] B. Widrow and M. E. Hoff, Jr., "Adaptive switching circuits," in *1960 IRE WESCON Conv. Rec.*, pt. 4, pp. 96–104.
[2] J. S. Koford and G. F. Groner, "The use of an adaptive threshold element to design a linear optimal pattern classifier," *IEEE Trans. Inform. Theory*, vol. IT-12, pp. 42–50, Jan. 1966.
[3] B. Widrow, "Adaptive filters," in *Aspects of Network and System Theory*, R. E. Kalman and N. DeClaris, Eds. New York: Holt, Rinehart, and Winston, 1971, pp. 563–587.
[4] N. J. Nilsson, *Learning Machines*. New York: McGraw-Hill, 1965.
[5] B. Widrow *et al.*, "Adaptive antenna systems," *Proc. IEEE*, vol. 55, pp. 2143–2159, Dec. 1967.
[6] J. G. Proakis, "An adaptive receiver for digital signaling through channels with intersymbol interference," *IEEE Trans. Inform. Theory*, vol. IT-15, pp. 484–497, July 1969.
[7] R. L. Riegler and R. T. Compton, Jr., "An adaptive array for interference rejection," *Proc. IEEE*, vol. 61, pp. 748–758, June 1973.

An Algorithm for the Inversion of Continued Fractions

V. V. BAPESWARA RAO AND V. K. AATRE

Abstract—A simple procedure for the inversion of a general continued fraction is presented.

Manuscript received July 22, 1974. This work was supported by the National Research Council of Canada under Grant A-7237.

The authors are with the Nova Scotia Technical College, Halifax, Nova Scotia, Canada.

The problem of inversion of a continued fraction is generally considered as the problem of construction of the Routh's array in the reverse order [1], [2]. Such an approach is not directly applicable when the continued fraction is terminated in a rational function. A procedure for the inversion of a continued fraction terminated in a rational function has been given by Chen and Chang [3]. Their method is based on the determination of the chain matrix of the relevant Cauer realization. The procedure is not attractive as it involves the processing of several polynomials. In this letter, the inversion of a general continued fraction (Cauer's third form) terminated in a rational function is achieved without evaluating the parameters of the chain matrix.

$$\begin{aligned} T(s) = 1\Big/\!\!\Big/\Big(h_1 + H_1 s + 1\Big/\!\!\Big/\Big(\frac{h_2}{s} + H_2 + 1\Big/\!\!\Big/\Big(\cdots \\ + 1\Big/\!\!\Big/\Big(h_{2k-1} + H_{2k-1}s + 1\Big/\!\!\Big/\Big(\frac{h_{2k}}{s} + H_{2k} + 1\Big/\!\!\Big/\Big(\cdots \\ + 1\Big/\!\!\Big/\Big(h_{2n-1} + H_{2n-1}s + 1\Big/\!\!\Big/\Big(\frac{h_{2n}}{s} + H_{2n} + G\Big)\Big)\Big)\cdots\Big)\Big)\cdots\Big)\Big)\Big) \end{aligned} \tag{1}$$

where $G = g_1(s)/g_2(s)$ is a rational function of s. (The argument s in the designation of the rational function is omitted for notational simplicity.) It is assumed that the last coefficients in the expansion are h_{2n} and H_{2n}. There is no loss of generality in this assumption as any given function can be reduced to this form by properly modifying G.

Let

$$\begin{aligned} T_k = p_k(s)/q_k(s) = 1\Big/\!\!\Big/\Big(h_{2k-1} + H_{2k-1}s + 1\Big/\!\!\Big/\Big(\frac{h_{2k}}{s} + H_{2k} \\ + 1\Big/\!\!\Big/\Big(h_{2k+1} + H_{2k+1}s + 1\Big/\!\!\Big/\Big(\cdots + 1\Big/\!\!\Big/\Big(h_{2n-1} \\ + H_{2n-1}s + 1\Big/\!\!\Big/\Big(\frac{h_{2n}}{s} + H_{2n} + G\Big)\Big)\Big)\cdots\Big)\Big)\Big). \end{aligned} \tag{2}$$

The rational functions $T_{k-1}, T_{k+1}, \cdots$, are similarly defined. Thus $T_1 = T(s)$ and $T_{n+1} = G$. With the notation in (2),

$$\begin{aligned} T_{k-1} &= \frac{p_{k-1}(s)}{q_{k-1}(s)} = 1\Big/\!\!\Big/\Big(h_{2k-1} + H_{2k-1}s + 1\Big/\!\!\Big/\Big(\frac{h_{2k}}{s} + H_{2k} + T_k\Big)\Big) \\ &= \frac{h_{2k}q_k(s) + sH_{2k}q_k(s) + sp_k(s)}{(h_{2k-1} + sH_{2k-1})\{h_{2k}q_k(s) + sH_{2k}q_k(s) + sp_k(s)\} + sq_k(s)}. \end{aligned} \tag{3}$$

Thus

$$p_{k-1}(s) = h_{2k}q_k(s) + sH_{2k}q_k(s) + sp_k(s)$$

and

$$q_{k-1}(s) = h_{2k-1}p_{k-1}(s) + sH_{2k-1}p_{k-1}(s) + sq_k(s). \tag{4}$$

These relations can be used to compute $p_1(s)$ and $q_1(s)$ successively from $g_1(s)$ and $g_2(s)$. The task can be mechanized by arranging the coefficients of the various polynomials in the form of a matrix C which may be called the coefficient matrix. The structure of C is shown in Fig. 1.

The number of rows of C is $(2n + 2)$ and the number of columns is $(2n + d + 1)$, where $d = \max\{\text{degree of } g_1(s), \text{degree of } g_2(s)\}$. With such an ordering, the polynomials $p_k(s)$ and $q_k(s)$ are given as

$$p_k(s) = \sum_{i=1}^{2n+d+1} c_{2n-2k+3,i}s^{i-1}$$

and

$$q_k(s) = \sum_{i=1}^{2n+d+1} c_{2n-2k+4,i}s^{i-1}. \tag{5}$$

The relationship between the elements of C, from (4), can be obtained as

$$c_{i,j} = h_{2n-i+3}c_{i-1,j} + H_{2n-i+3}c_{i-1,j-1} + c_{i-2,j-1}. \tag{6}$$