# Efficient Lattice Realizations of Adaptive IIR Algorithms

Sergio Lima Netto and Panajotis Agathoklis

*Abstract*—**Computationally efficient adaptive IIR-filter algorithms are presented based on lattice realizations allowing the adaptive filter stability to be easily monitored. New simplified recursive-in-order equations relating the parameters of the direct-form realization with the ones of two lattice realizations are presented. These equations lead to a simplified technique to compute the regressor vector and to a general method to implement any adaptive IIR algorithm using lattice realization. Results indicate that the proposed lattice-based algorithms converge to a set of parameters that realize the same transfer function as the corresponding direct-form algorithms.**

*Index Terms*—**Adaptive IIR filters, lattice structure.**

## I. Introduction

Adaptive IIR filters constitute a potential alternative to adaptive FIR filters as they are suitable for modeling real systems with sharp resonances using significantly fewer coefficients. Standard adaptive IIR filter algorithms are commonly presented in the literature based on the direct-form realization to obtain a simpler understanding of the nature of the respective algorithm as well as of its convergence properties. The direct-form realization, however, is not suitable for most practical implementations because it does not allow an efficient on-line stability testing, which is required by several adaptation algorithms to avoid filter instability during the convergence process. Consequently, several alternative structures have been considered for the implementation of adaptive IIR filter algorithms.

The lattice realization [2], [3], [6] is an example of a filter structure, the stability of which can be easily ensured in real time, making this structure well suited for adaptive IIR filtering. Initial attempts to use the lattice structure have led to computationally complex adaptive algorithms [5], [11]. Subsequent efforts to simplify [1] and accelerate [14] these algorithms have led to parameter drift during the convergence process or further increase in the computational complexity. More recently, in [8], [12], and [13], lattice-based adaptive IIR algorithms with guaranteed convergence and better computational complexity have been proposed.

This correspondence deals with the lattice-based implementation of adaptive IIR filter algorithms using recursive-in-order equations for the numerator and denominator polynomials of the adaptive filter transfer function, as opposed to the matrix approach used in [8]. This results in simpler recursion equations, and it also allows the implementation of adaptive algorithms from the equation error (EE) family of algorithms (e.g., the bias-remedy EE [7] and the composite algorithms of [9] and [10].) The computationally efficient algorithms are then obtained through a simplification in the gradient vector first used in [13]. In that sense, although the proposed approach presents computational complexity comparable with the methods given in [8] and [12], it is said to be more general as it can be used to implement any given adaptive IIR-filter algorithm using lattice structures.

The correspondence is organized as follows. In the next section, the direct-form EE algorithm is given following a general framework for the description of adaptation algorithms. Later, the two-multiplier lattice structure is presented along with a new technique for finding a lattice realization of a given transfer function. In Section IV, using that relationship, we present a computationally efficient implementation of the EE adaptation algorithm based on the two-multiplier lattice realization. Additionally, directions are given to extend the proposed method to other IIR adaptation algorithms and to the normalized lattice realization. Computer simulations are then included to demonstrate the validity and the usefulness of the proposed techniques.

Manuscript received November 2, 1995; revised March 17, 1997. This work was supported by CAPES-Ministry of Education Brazil and Micronet, Canada. The associate editor coordinating the review of this paper and approving it for publication was Dr. José Carlos M. Bermudez.

S. L. Netto was at the Department of Electrical and Computer Engineering, University of Victoria, Victoria, B.C., Canada. He is now with the Departamento de Eletrônica, Universidade Federal de Rio de Janeiro, Rio de Janeiro, Brazil (e-mail: sergioln@coe.ufrj.br).

P. Agathoklis is with the Department of Electrical and Computer Engineering, University of Victoria, Victoria, B.C., Canada (e-mail: pan@sirius.uvic.ca).

Publisher Item Identifier S 1053-587X(98)00520-0.

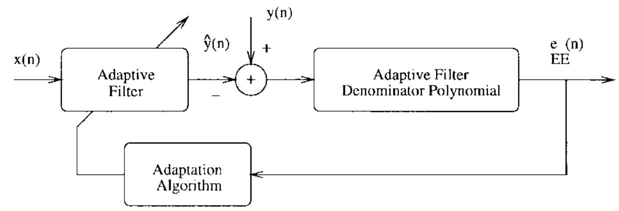


Fig. 1. Block diagram of the equation-error scheme.

## II. Adaptive IIR Algorithms

Fig. 1 depicts the basic block diagram of a general adaptive system. In this figure

- $x(n)$ input signal;
- $y(n)$ desired output or reference signal;
- $\hat{y}(n)$ adaptive filter output signal;
- $e_{\mathrm{EE}}(n)$ so-called EE signal.

Using the direct-form realization, the adaptive filter is described by

$$\hat{y}(n) = \left[\frac{\hat{B}(q,n)}{\hat{A}(q,n)}\right]\{x(n)\} \tag{1}$$

with $\hat{A}(q,n) = 1 + \sum_{i=1}^{n_{\hat{a}}} \hat{a}_i(n)q^{-i}$ and $\hat{B}(q,n) = \sum_{j=0}^{n_{\hat{a}}} \hat{b}_j(n)q^{-j}$, where $q^{-i}$ is the delay operator defined by $q^{-i}\{x(n)\} = x(n-i)$.

The basic form of a general adaptive filtering algorithm can be written as

$$\hat{\boldsymbol{\theta}}(n+1) = \hat{\boldsymbol{\theta}}(n) + \mu(n)e(n)\hat{\phi}(n) \tag{2}$$

where

- $\mu(n)$ gain factor that can be a matrix or a scalar;
- $e(n)$ estimation error;
- $\hat{\phi}(n)$ regressor or information vector associated to the respective adaptation algorithm.

In addition, $\hat{\boldsymbol{\theta}}(n)$ is the adaptive filter coefficient vector that, for the direct structure characterized above, is defined as[1]

$$\hat{\boldsymbol{\theta}}_d(n) = \left[\hat{a}_1(n) \cdots \hat{a}_{n_{\hat{a}}}(n)\ \hat{b}_0(n) \cdots \hat{b}_{n_{\hat{b}}}(n)\right]^T. \tag{3}$$

Following (2), the EE algorithm that minimizes the mean square equation error is described by

$$\begin{aligned} e(n) \equiv e_{\mathrm{EE}}(n) &= \hat{A}(q, n)\{e_{OE}(n)\} \\ &= \hat{A}(q, n)\{y(n)\} - \hat{B}(q, n)\{x(n)\} \qquad (4a) \\ \hat{\phi}(n) \equiv \hat{\phi}_{\mathrm{EE}}(n) & \\ &= [-y(n-1) \cdots -y(n-n_{\hat{a}}) \quad x(n) \cdots x(n-n_{\hat{b}})]^T. \qquad (4b) \end{aligned}$$

## III. The Two-Multiplier Tapped Lattice IIR Realization

As described in [2], a rational transfer function of the form

$$H(z) = \frac{B_N(z)}{A_N(z)} = \frac{b_0 + b_1 z^{-1} + \cdots + b_N z^{-N}}{1 + a_1 z^{-1} + \cdots + a_N z^{-N}} \tag{5}$$

can be implemented using an alternative set of parameters $\hat{\boldsymbol{\theta}}_{2\ell} = [k_1 \cdots k_N \quad h_0 \cdots h_N]^T$, corresponding to the two-multiplier lattice realization obtained from

$$\begin{aligned} A_{m-1}(z) &= [A_m(z) - k_m A_m(z^{-1}) z^{-m}]/(1 - k_m^2) \qquad (6a) \\ B_{m-1}(z) &= B_m(z) - A_m(z^{-1}) z^{-m} h_m \\ & \qquad m = N, \cdots, 1 \qquad (6b) \end{aligned}$$

where the auxiliary polynomials $A_m(z)$ and $B_m(z)$ are defined as

$$\begin{aligned} A_m(z) &= a_{m,0} + a_{m,1} z^{-1} + \cdots + a_{m,m} z^{-m} \qquad (7a) \\ B_m(z) &= b_{m,0} + b_{m,1} z^{-1} + \cdots + b_{m,m} z^{-m} \\ & \qquad m = N, \cdots, 1 \qquad (7b) \end{aligned}$$

with $a_{m,0} = 1$ such that $k_m = a_{m,m}$, $h_m = b_{m,m}$ for $m = N, \cdots, 1$, and $h_0 = b_{0,0}$.

Using the $\hat{\boldsymbol{\theta}}_{2\ell}$ parameters, the output signal $y(n)$ can be calculated by

$$\begin{aligned} F_i(n) &= F_{i+1}(n) - k_{i+1} G_i(n-1) \\ & \qquad i = N-1, \cdots, 0 \qquad (8a) \\ G_j(n) &= G_{j-1}(n-1) + k_j F_{j-1}(n) \\ & \qquad j = 1, \cdots, N \qquad (8b) \\ y(n) &= \sum_{j=0}^{N} h_j G_j(n) \qquad (8c) \end{aligned}$$

with $F_N(n) = x(n)$ and $G_0(n) = F_0(n)$. An alternative approach to implement the $H(z)$ transfer function while still using the $\hat{\boldsymbol{\theta}}_{2\ell}$ coefficients results from the relationships given in the following lemma.

*Lemma 1:* Consider the $A_m(z)$ and $B_m(z)$ polynomials as given in (7) and the parameter vector $\hat{\boldsymbol{\theta}}_{2\ell}$. Then, the recursive-in-order equations

$$\begin{aligned} A_m(z) &= A_{m-1}(z) + \frac{k_m}{k_{m-1}} \\ & \quad \cdot [A_{m-1}(z) - (1 - k_{m-1}^2) A_{m-2}(z)] z^{-1} \qquad (9a) \\ B_m(z) &= B_{m-1}(z) + \frac{h_m}{k_m} [A_m(z) - (1 - k_m^2) A_{m-1}(z)] \\ & \qquad m = 2, \cdots, N \qquad (9b) \end{aligned}$$

[1]The subscripts $d$ and $\ell$ will be used throughout the text to associate a given variable, respectively, to the direct-form or lattice realizations. More specifically, the subscripts $2\ell$ and $4\ell$ will refer to the two-multiplier and normalized lattice structures, respectively.

hold, with $A_0(z) = 1$, $A_1(z) = 1 + k_1 z^{-1}$, and $B_1(z) = (h_0 + h_1 k_1) + h_1 z^{-1}$.

In addition, with $A'_m(z) = A_m(z) - 1 = a_1 z^{-1} + \cdots + a_m z^{-m}$, (9a) leads to

$$\begin{aligned} A'_m(z) &= A'_{m-1}(z) + \frac{k_m}{k_{m-1}} \\ & \quad \cdot [A_{m-1}(z) - (1 - k_{m-1}^2) A_{m-2}(z)] z^{-1} \qquad (10) \end{aligned}$$

for the same initial conditions for $A_m(z)$ and values of $m$ as before.

Proof for this lemma is obtained from simple algebraic manipulation of (6) for the filter described by (5). In the next section, we indicate how the recursions given in Lemma 1 can be used to implement lattice-based algorithms for adaptive IIR filters.

## IV. Efficient Lattice-Based Algorithms for Adaptive IIR Filters

Equation (2) indicates that the implementation of an adaptive IIR algorithm requires the calculation of a residual error signal and of an information vector. Those basic procedures fundamentally consist of processing present or past samples of $x(n)$, $y(n)$, $\hat{y}(n)$, or any other auxiliary signal, with the available numerator or denominator polynomials of the adaptive filter. Hence, in order to derive lattice IIR algorithms, it is then necessary to find a possible way to perform any of that additional processing based solely on the coefficients $\hat{\boldsymbol{\theta}}_{2\ell}$. For that, we must first generalize recursions (9) and (10) to the time-varying coefficient case using the so-called small-step approximation [4].

The proposed method to derive lattice-based adaptation algorithms is then accomplished by implementing the direct-form version of the algorithm, using, however, the lattice set of coefficients $\boldsymbol{\theta}_{2\ell}$ through the time-varying extensions of the recursions given in Lemma 1. In that way, we have that $\hat{\phi}_\ell(n) = \hat{\phi}_d(n)$, and lattice-based algorithms are efficiently implemented requiring $O(N)$ multiplication/division operations, as opposed to the $O(N^2)$ operations required by earlier lattice adaptation algorithms [5], [11]. Thus, the algorithms proposed here, along with the ones presented in [8], [12], and [13], present similar computational complexity to their equivalent direct-form counterparts with the additional property of allowing real-time pole monitoring.

*Property 1:* The proposed lattice version of any given adaptive IIR-filter algorithm is equivalent to its direct-form standard implementation in the sense that both methods present sets of stationary points corresponding to the same input–output descriptions of the adaptive filter.

This result was first verified in [12] based on the fact that the stationary points of an adaptive algorithm are the solutions of

$$E[e(n)\hat{\phi}(n)] = \mathbf{0}. \tag{11}$$

For equivalent direct-form and two-multiplier lattice realizations, the residual errors are equal, and using the proposed simplification, the corresponding regression vectors are also identical. Consequently, one has that

$$E[e_d(n)\hat{\phi}_d(n)] = \mathbf{0} \iff E[e_\ell(n)\hat{\phi}_\ell(n)] = \mathbf{0}. \tag{12}$$

Property 1 indicates the steady-state equivalence between the proposed lattice approach and the corresponding direct-form algorithm. The transient part of the adaptation process, on the other hand, is not equivalent to these two schemes, and the convergence conditions for the proposed method are still an open problem.

TABLE I
SUMMARY OF THE EFFICIENT TWO-MULTIPLIER LATTICE EE ADAPTIVE ALGORITHM

**step 0** - Initialize:

The convergence parameter $\mu$;

The adaptive filter coefficient vector $\hat{\boldsymbol{\theta}}_{2\ell} = \left[\hat{k}_1(n) \ \ldots \ \hat{k}_N(n) \ \ \hat{h}_0(n) \ \ldots \ \hat{h}_N(n)\right]^T$;

The auxiliary vectors:

$$\begin{bmatrix} \hat{A}_0(q,n)\{x(n-1)\} \\ \vdots \\ \hat{A}_{N-1}(q,n)\{x(n-1)\} \end{bmatrix} \text{ and } \begin{bmatrix} \hat{A}_0(q,n)\{y(n-1)\} \\ \vdots \\ \hat{A}_{N-1}(q,n)\{y(n-1)\} \end{bmatrix}.$$

**step 1** - Compute:

$$\hat{A}_m(q,n)\{y(n)\} = \hat{A}_{m-1}(q,n)\{y(n)\} + \frac{\hat{k}_m(n)}{\hat{k}_{m-1}(n)}\left[\hat{A}_{m-1}(q,n)\{y(n-1)\} - (1-\hat{k}^2_{m-1}(n))\hat{A}_{m-2}(q,n)\{y(n-1)\}\right],$$

for $m = 2, \ldots, N$; with $\hat{A}_1(q,n)\{y(n)\} = y(n) + \hat{k}_1(n)y(n-1)$;

$$\hat{A}_m(q,n)\{x(n)\} = \hat{A}_{m-1}(q,n)\{x(n)\} + \frac{\hat{k}_m(n)}{\hat{k}_{m-1}(n)}\left[\hat{A}_{m-1}(q,n)\{x(n-1)\} - (1-\hat{k}^2_{m-1}(n))\hat{A}_{m-2}(q,n)\{x(n-1)\}\right],$$

for $m = 2, \ldots, N$; with $\hat{A}_1(q,n)\{x(n)\} = x(n) + \hat{k}_1(n)x(n-1)$;

$$\hat{B}_m(q,n)\{x(n)\} = \hat{B}_{m-1}(q,n)\{x(n)\} + \frac{\hat{h}_m(n)}{\hat{k}_m(n)}\left[\hat{A}_m(q,n)\{x(n)\} - (1-\hat{k}^2_m(n))\hat{A}_{m-1}(q,n)\{x(n)\}\right],$$

for $m = 2, \ldots, N$; with $\hat{B}_1(q,n)\{x(n)\} = (\hat{h}_0(n) + \hat{h}_1(n)\hat{k}_1(n))x(n) + \hat{h}_1(n)x(n-1)$.

**step 2** - Obtain: $e_{EE}(n) = \hat{A}_N(q,n)\{y(n)\} - \hat{B}_N(q,n)\{x(n)\}$.

**step 3** - Form: $\hat{\boldsymbol{\phi}}_{EE_{2\ell}}(n) = [-y(n-1)\ldots - y(n-N) \ \ x(n)\ldots x(n-N)]^T$.

**step 4** - Update:

The adaptive filter coefficient vector: $\hat{\boldsymbol{\theta}}_{2\ell}(n+1) = \hat{\boldsymbol{\theta}}_{2\ell}(n) + \mu e_{EE}(n)\hat{\boldsymbol{\phi}}_{EE_{2\ell}}(n)$;

The time counter $n$;

The convergence parameter $\mu$ (optional);

The auxiliary vectors defined in **step 0**.

Return to **step 1**.

Table I includes an easy-to-follow routine describing the computationally efficient two-multiplier lattice version of the EE algorithm, for which the following comments apply.

1) To avoid division problems in **step 1**, the coefficients $\hat{k}_m(n)$ should be tested against getting too close to zero. In those cases, the recursions shown in Table I should be skipped without causing any problem to the algorithm's convergence process.
2) The adaptive filter coefficients in **step 4** can also be updating by a quasi-Newton type algorithm similar to the one described in [14] for the direct-form realization.
3) The stability monitoring in **step 5** must check if the absolute value of any coefficient $\hat{k}_m(n)$ becomes greater or equal to unity. If that happens, the coefficient must be stabilized by forcing its value to be inside the open interval $\hat{k}_m(n) \in (-1, 1)$ [3].

*Example 1:* Consider the system identification example of [11], where the plant is defined as

$$H(q) = \frac{0.0154 + 0.0462q^{-1} + 0.0462q^{-2} + 0.0154q^{-3}}{1 - 1.99q^{-1} + 1.572q^{-2} - 0.4583q^{-3}} \tag{13}$$

which yields the direct-form and two-multiplier coefficient vectors, respectively, given by

$$\boldsymbol{\theta}_d = \begin{bmatrix} a_1 \\ a_2 \\ a_3 \\ b_0 \\ b_1 \\ b_2 \\ b_3 \end{bmatrix} = \begin{bmatrix} -1.9900 \\ 1.5720 \\ -0.4583 \\ 0.0154 \\ 0.0462 \\ 0.0462 \\ 0.0154 \end{bmatrix}; \quad \boldsymbol{\theta}_{2\ell} = \begin{bmatrix} k_1 \\ k_2 \\ k_3 \\ h_0 \\ h_1 \\ h_2 \\ h_3 \end{bmatrix} = \begin{bmatrix} -0.8756 \\ 0.8355 \\ -0.4583 \\ 0.0856 \\ 0.1455 \\ 0.0768 \\ 0.0154 \end{bmatrix}. \tag{14}$$

Assume an adaptive filter with $N = 3$, and let the input signal be a Gaussian white noise with zero mean and unit variance. Consider the cases with and without a perturbation signal consisting of a zero-mean white noise of variance $\sigma_v^2 = 0.007$, corresponding to a signal-to-noise ratio of $\sigma_y^2/\sigma_v^2 = 15.1$ dB and infinity, respectively. Fig. 2 depicts the convergence of the coefficient vector using the EE algorithm realized by the two-multiplier lattice structure for the two cases. It can be noted that in the presence of perturbation, the adaptive filter converged to a biased solution with respect to the optimal one indicated by doted lines. For the second case the optimal solution was reached as expected.

Although this correspondence focuses on lattice realizations of the EE algorithm, extension of the proposed method to other adaptive IIR-filtering algorithms is easily accomplished. For that, one must only compute the corresponding error signal and

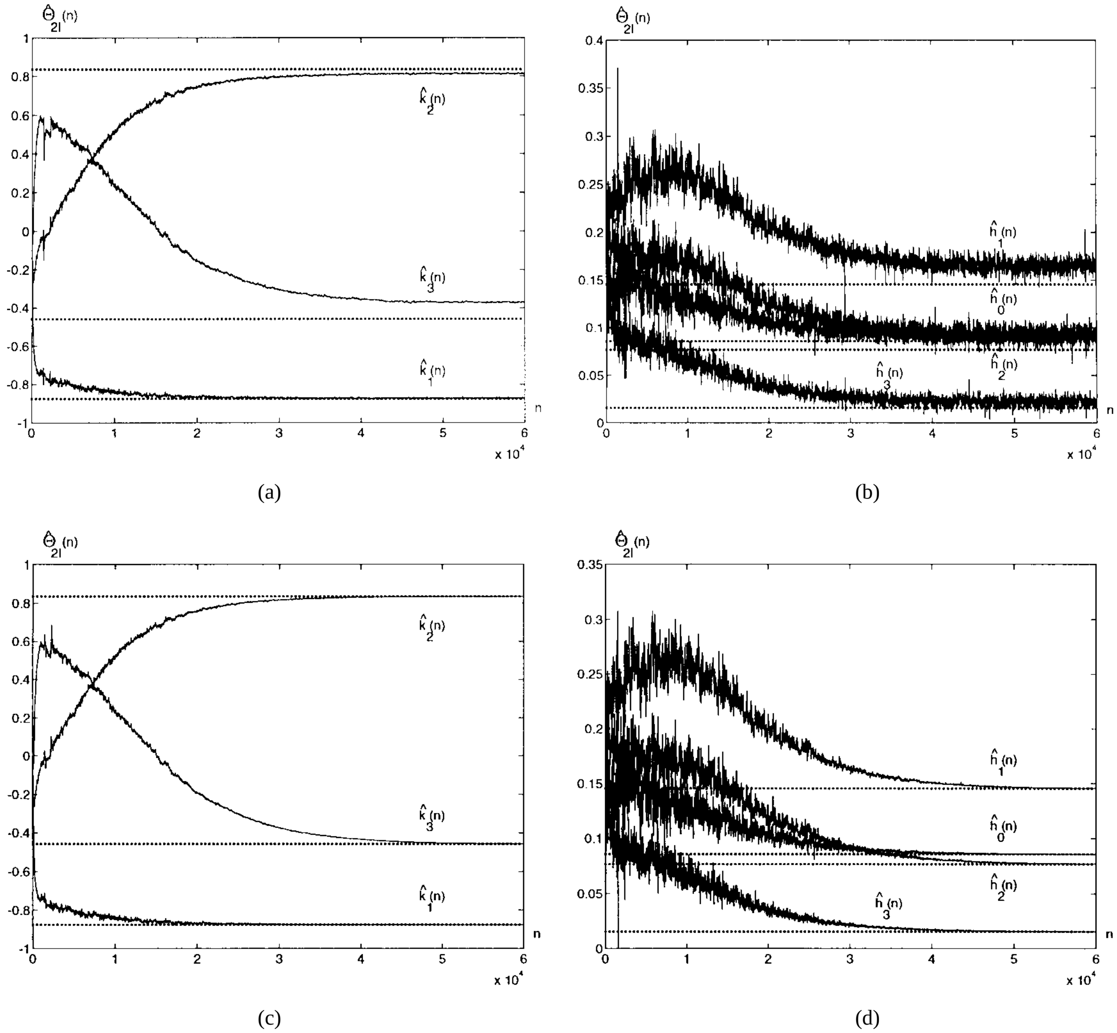


Fig. 2. Lattice adaptive convergence, $\mu_\ell = 0.06$. With perturbation noise: (a) Two-multiplier reflection coefficients and (b) two-multiplier forward coefficients. Without perturbation noise: (c) Two-multiplier reflection coefficients and (d) two-multiplier forward coefficients.

regressor vector based on their original direct-form definitions by using the time-varying extensions of the recursion-in-order equations (9) and (10), as it was illustrated here for the EE algorithm.

### *A. Normalized-Lattice Adaptive IIR Algorithms*

The extension of adaptive IIR algorithms from the two-multiplier lattice to the normalized lattice realization follows naturally from the relationships existing between the coefficients of these two structures. Indeed, the normalized lattice structure has the set of coefficients $\hat{\boldsymbol{\theta}}_{4\ell} = [\phi_1 \cdots \phi_N \quad h'_0 \cdots h'_N]^T$ that are related to the entries of $\hat{\boldsymbol{\theta}}_{2\ell}$ by [3]

$$\sin\,\phi_i = k_i; \qquad i = 1, \cdots, N \tag{15a}$$

$$h'_j = \frac{h_j}{\pi_j}; \qquad j = 0, \cdots, N \tag{15b}$$

where the parameters $\pi_j$ are given by

$$\pi_{j-1} = \pi_j \, \cos\,\phi_{j-1}; \qquad j = N-1, \cdots, 1 \tag{16}$$

with $\pi_N = 1$. Using those relationships with the equations given in Lemma 1, the following holds.

*Lemma 2:* Consider the $A_m(z)$ and $B_m(z)$ as defined in (7). For the normalized lattice realization, those polynomials may be recursively computed as

$$A_m(z) = A_{m-1}(z) + \frac{\sin\,\phi_m}{\sin\,\phi_{m-1}} \cdot [A_{m-1}(z) - \cos^2\,\phi_{m-1} A_{m-2}(z)] z^{-1} \tag{17a}$$

$$B_m(z) = B_{m-1}(z)\,\cos\,\phi_m + \frac{h'_m}{\sin\,\phi_m} \cdot [A_m(z) - \cos^2\,\phi_m A_{m-1}(z)] \qquad m = 2, \cdots, N \tag{17b}$$

with $A_0(z) = 1$, $A_1(z) = 1 + \sin\,\phi_1 z^{-1}$, and $B_1(z) = (h'_0\,\cos\,\phi_1 + h'_1\,\sin\,\phi_1) + h'_1 z^{-1}$.

As a consequence of (16a), with $A'_m(z)$ as before, one has

$$A'_m(z) = A'_{m-1}(z) + \frac{\sin\,\phi_m}{\sin\,\phi_{m-1}} \cdot [A_{m-1}(z) - \cos^2\,\phi_{m-1} A_{m-2}(z)] z^{-1}. \tag{18}$$

As shown in [12], exponential stability of the time-varying normalized lattice realization is guaranteed as long as the reflection coefficients satisfy $|\phi_i(n)| \le \pi/2 - \delta$ for all $n$ and a positive $\delta$.

## V. Conclusions

In this correspondence, relationships of the transfer function polynomials of the direct-form and two lattice realizations were introduced. It was shown that these equations yield computationally efficient implementation of lattice-based algorithms for adaptive IIR filters, including members of the equation error family of algorithms, requiring $O(N)$ multiplications per iteration, where $N$ is the filter order. The proposed lattice-based implementation led to a set of parameters that realize identical transfer functions to the ones obtained by corresponding direct-form algorithms.

## Acknowledgment

The authors would like to thank the reviewers for all their helpful comments.

## References

[1] I. L. Ayala, "On a new adaptive lattice algorithm for recursive filters," *IEEE Trans. Acoust., Speech, Signal Processing,* vol. ASSP-30, pp. 316–319, Apr. 1982.

[2] A. H. Gray, Jr. and J. D. Markel, "Digital lattice and ladder filter synthesis," *IEEE Trans. Audio Electroacoust.,* vol. 21, pp. 492–500, Dec. 1973.

[3] ——, "A normalized digital filter structure," *IEEE Trans. Acoust., Speech, Signal Processing,* vol. ASSP-23, pp. 268–277, June 1975.

[4] S. Horvath, Jr., "A new adaptive recursive LMS filter," *Digital Signal Processing,* V. Cappellini and A. G. Constantinides, Eds. New York: Academic, 1980, pp. 21–26.

[5] ——, "Lattice form adaptive recursive digital filters: Algorithms and applications," in *Proc. Int. Symp. Circuits Syst.,* Houston, TX, Apr. 1980, pp. 128–133.

[6] F. Itakura and S. Saito, "Digital filtering techniques for speech analysis and synthesis," in *Proc. 7th Int. Cong. Acoustics,* Budapest, Hungary, 1971, pp. 261–264.

[7] J.-N. Lin and R. Unbehauen, "Bias-remedy least mean square equation error algorithm for IIR parameter recursive estimation," *IEEE Trans. Signal Processing,* vol. 40, pp. 62–69, Jan. 1992.

[8] K. X. Miao, H. Fan, and M. Doroslovački, "Cascade lattice IIR adaptive filters," *IEEE Trans. Signal Processing,* vol. 42, pp. 721–742, Apr. 1994.

[9] S. L. Netto and P. S. R. Diniz, "Composite algorithms for adaptive IIR filtering," *IEE Electron. Lett.,* vol. 28, no. 9, pp. 886–888, Apr. 1992.

[10] S. L. Netto and P. Agathoklis, "A new composite adaptive IIR algorithm," in *Proc. 28th Asilomar Conf. Signals, Systems, Computers,* Pacific Grove, CA, Oct./Nov. 1994, pp. 1506–1510.

[11] D. Parikh, N. Ahmed, and S. D. Stearns, "An adaptive lattice algorithm for recursive filters," *IEEE Trans. Acoust., Speech, Signal Processing,* vol. ASSP-28, pp. 110–112, Feb. 1980.

[12] P. A. Regalia, "Stable and efficient lattice algorithms for adaptive IIR filtering," *IEEE Trans. Signal Processing,* vol. 40, pp. 375–388, Feb. 1992.

[13] J. A. Rodríguez-Fonollosa and E. Masgrau, "Simplified gradient calculation in adaptive IIR lattice filters," *IEEE Trans. Signal Processing,* vol. 39, pp. 1702–1705, July 1991.

[14] J. J. Shynk, "On lattice-form algorithms for adaptive IIR filtering," in *Proc. Int. Conf. Acoustics, Speech, Signal Processing,* New York, Apr. 1988, pp. 1554–1557.

# Adaptive Fractionally Spaced Blind CMA Equalization: Excess MSE

I. Fijalkow, C. E. Manlove, and C. R. Johnson, Jr.

***Abstract*— The performance of the constant modulus algorithm (CMA)—a reference algorithm for adaptive blind equalization—is studied in terms of the excess mean square error (EMSE) due to the nonvanishing step size of the gradient descent algorithm. An analytical approximation of EMSE is provided, emphasizing the effect of the constellation size and resulting in design guidelines.**

***Index Terms*—Adaptive blind equalization, excess mean square error, multichannel equalization.**

## I. Introduction

Digital communication is subject to intersymbol interference (ISI) due to nonideal pulse shaping, multipath propagation, and residual clock or carrier phase error. ISI is more severe when the channel dispersion time (or channel time span) cannot be neglected with respect to the input signal symbol time duration, thereby making its removal all the more crucial. Traditionally, channel equalization (i.e., input sequence extraction directly from the received signal) and identification (with input sequence recovery, e.g., by Wiener filtering of the observed signal using the channel estimate) are performed using a training sequence. However, in many applications, either the training sequence is unavailable, or the bandwidth occupied by the training sequence is to be spared for input carrying information. Consequently, one pursues *blind equalization*, i.e., without training nor any *a priori* knowledge of the channel dynamics. Due to its potential benefits, blind equalization has become an important topic in digital communications. Blind methods use the received signal sequence and some *a priori* knowledge of the input sequence statistics. Nonminimum phase channel equalization was performed using methods based on high-order statistics or other nonlinearities that are effective only with non-Gaussianly distributed input sequences [4]. Contrast-based methods (see [13], for example) or adaptive Bussgang algorithms (see [1]) have been proposed and studied over the last ten years. In this correspondence, we study the most popular adaptive blind equalization Bussgang algorithm (the Godard algorithm [7]) or constant modulus algorithm (CMA) [15] in the context of nonconstant modulus data with spatio-temporal diversity.

The constant modulus (CM) criterion minima have been proven to achieve zero-forcing fractionally spaced equalization (see [5] and [11] for a simple algebraic proof) under various ideal conditions, including the absence of noise. Even under the presence of a small amount of additive channel noise, CM minima were proved to

Manuscript received January 27, 1995; revised June 17, 1997. This work was supported by the French Ministére des Affaires Etrangéres (bourse Lavoisier), Applied Signal Technology, Inc., and NSF Grant MIP-9509011. The associate editor coordinating the review of this paper and approving it for publication was Dr. Pierre Duhammel.

I. Fijalkow was with the School of Electrical Engineering, Cornell University, Ithaca, NY 14853 USA. She is now with ENSEA de Cergy, Groupe ETIS, Cergy-Pontoise, France (e-mail: fijalkow@ensea.fr).

C. E. Manlove was with School of Electrical Engineering, Cornell University, Ithaca, NY 14853 USA. She is now with the Applied Physics Laboratory, Johns Hopkins University, Baltimore, MD 21218 USA.

C. R. Johnson is with the School of Elecricial Engineering, Cornell University, Ithaca, NY 14853 USA (e-mail: johnson@anise.ee.cornell.edu).

Publisher Item Identifier S 1053-587X(98)00505-4.