Composite Algorithms for Adaptive IIR Filtering Using Lattice Realization

S. L. Netto and P. Agathoklis
Department of Electrical and Computer Engineering
University of Victoria - P.O. Box 3055 - MS 8610
Victoria - B.C. - Canada - V8W 3P6
snetto@sirius.uvic.ca and pan@sirius.uvic.ca

Abstract: **Adaptive IIR algorithms are implemented based on lattice realizations allowing the adaptive filter poles to be monitored in real-time. New simplified recursive-in-order equations are presented relating the parameters of the direct-form realization to the ones of the two-multiplier IIR lattice realization. Those equations yield a general method to implement any adaptive IIR algorithm including the members of the equation error family of algorithms. Based on these new techniques, computationally efficient algorithms requiring $O(N)$ multiplications are obtained for the lattice structure. Simulations are included to demonstrate the usefulness and validity of the proposed methods.**

I. INTRODUCTION

Adaptive IIR filters are a potential alternative to adaptive FIR filters as they are able to model real systems with sharp resonances using significantly less coefficients. Standard adaptive IIR 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 practical implementations of adaptive filters because it does not allow an efficient on-line pole monitoring as required by several adaptive IIR algorithms to avoid instability of the adaptive filter. Consequently, several alternative structures have been considered for the implementation of adaptive IIR algorithms.

The lattice realization [1], [2] is an example of a filter structure the stability of which can be ensured in real time making this structure well suited for adaptive IIR filtering. The relationships between the solutions of adaptive algorithms based on the lattice and the direct-form realizations were first studied by Nayeri in [3]. In [3], it was proven that the convergence points of an adaptive algorithm using the lattice structure present an one-to-one correspondence with the solutions of the direct-form version of the same algorithm. This property, further motivated the idea of using the lattice structure as an efficient and equivalent alternative realization for adaptive IIR filters. Initial attempts of applying lattice structure to adaptive IIR filtering [4], [5] have lead to computationally complex adaptive algorithms. New more efficient lattice-based adaptive IIR algorithms have been presented in the literature [6]-[8] recently. However, those algorithms do not include the equation error family of algorithms.

In this paper, new methods to implement adaptive algorithms using lattice realizations are presented. Due to generality of those new techniques, lattice-based versions can be obtained for any currently known adaptive IIR algorithm, including the members of the equation error family of algorithms.

The paper is organized as follows: In the next section, the two-multiplier lattice form is presented along with a new technique for finding a similar lattice realization of a given transfer function. In section III, an efficient form to implement lattice-based adaptive IIR algorithms is discussed. A composite adaptive algorithm is implemented using the proposed techniques and used in computer simulations to verify the performance of the proposed method in practical situations.

*This work was supported by CAPES - Ministry of Education (Brazil) and Micronet (Canada).

II. THE TWO-MULTIPLIER IIR LATTICE REALIZATION

As described in [1], a rational transfer function of the form

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

can be implemented using an alternative set of parameters $\boldsymbol{\theta}_{2\ell} = [k_1 \; \ldots \; k_N \;\; h_0 \; \ldots \; h_N]^T$ [1] obtained from the following set of equations [1]:

$$A_{m-1}(z) = \left[A_m(z) - k_m A_m(z^{-1}) z^{-m}\right] / (1 - k_m^2) \quad (2a)$$

$$B_{m-1}(z) = B_m(z) - A_m(z^{-1}) z^{-m} h_m; \; m = N, \ldots, 1 \quad (2b)$$

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

$$A_m(z) = a_{m,0} + a_{m,1} z^{-1} + \ldots + a_{m,m} z^{-m} \quad (3a)$$

$$B_m(z) = b_{m,0} + b_{m,1} z^{-1} + \ldots + b_{m,m} z^{-m}; \; m = N, \ldots, 1 \quad (3b)$$

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

with $a_{m,0} = 1$.The $\hat{\theta}_{2\ell}$ coefficients are then obtained by $k_m = a_{m,m}$ and $h_m = b_{m,m}$, for $m = N, \ldots, 1$, and also $h_0 = b_{0,0}$.

Using the $\theta_{2\ell}$ coefficients then, the output signal, $y(n)$, of the corresponding two-multiplier lattice filter to an input signal $x(n)$, can be calculated by the following set of equations:

$$F_i(n)=F_{i+1}(n) - k_{i+1}G_i(n-1);\ i = N-1, \ldots, 0 \quad \text{(4a)}$$

$$G_j(n)=G_{j-1}(n-1) + k_j F_{j-1}(n);\ j = 1, \ldots, N \quad \text{(4b)}$$

$$y(n)=\sum_{j=0}^{N} h_j G_j(n) \quad \text{(4c)}$$

with $F_N(n) = x(n)$ and $G_0(n) = F_0(n)$.

An alternative approach to implement the $H(z)$ transfer function still using the $\theta_{2\ell}$ coefficients is obtained based on the relationships given in the following lemma:

Lemma 1: Consider the $A_m(z)$ and $B_m(z)$ polynomials for $m = N, \ldots, 1$ as given in (3) and the parameter vector $\theta_{2\ell}$. Then, the following recursive-in-order equations hold:

$$A_m(z)=A_{m-1}(z) + \frac{k_m}{k_{m-1}} \cdot \left[A_{m-1}(z)-(1-k_{m-1}^2)A_{m-2}(z)\right] z^{-1} \quad \text{(5a)}$$

$$B_m(z)=B_{m-1}(z) + \frac{h_m}{k_m} \cdot \left[A_m(z)-(1-k_m^2)A_{m-1}(z)\right];\ m = 2, \ldots, N \quad \text{(5b)}$$

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

The proof of of this lemma is obtained from algebraic manipulation of equations (2) for the filter described by (1). The initial conditions for $m = 0, 1$ are obtained by simple algebraic calculations. It is interesting to note that equations (5) include solely causal polynomials while equations (2) include also noncausal terms. An important consequence of equation (5a) is that:

$$A'_m(z)=A'_{m-1}(z) + \frac{k_m}{k_{m-1}} \cdot \left[A_{m-1}(z)-(1-k_{m-1}^2)A_{m-2}(z)\right] z^{-1} \quad \text{(6)}$$

for the same initial conditions for $A_m(z)$ and values of m as before, where the auxiliary polynomial $A'_m(z)$ is defined as:

$$A'_m(z) = A_m(z) - 1 = a_1 z^{-1} + \ldots + a_m z^{-m} \quad \text{(7)}$$

Based on the above given recursions, the output $y(n)$ of an IIR filter $H(z)$ to an input signal $x(n)$ can then be obtained by rewriting the input/output relationship in the time domain as:

$$y(n) = B_N(q)\{x(n)\} - A'_N(q)\{y(n)\} \quad \text{(8)}$$

For the computation of the above equation using lemma 1, it is required that $B_N(q)\{x(n)\}$ is obtained using equation (5b) recursively with $m = 2, \ldots, N$ and analogously $A'_N(q)\{y(n)\}$ using equation (6). It can be easily seen that the computation effort required to compute the output signal $y(n)$ via the above equation (8) is higher that the one required by the standard form given by (4). In the next section, however, it will be indicated how the recursions given in lemma 1 can be used in the efficient implementation of lattice-based adaptive IIR algorithms.

III. LATTICE-BASED ADAPTIVE IIR ALGORITHMS

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

$$\hat{\theta}(n+1) = \hat{\theta}(n) + \mu(n)e(n)\hat{\phi}(n) \quad \text{(9)}$$

where $\hat{\theta}(n)$ is the adaptive filter coefficient vector, $\mu(n)$ is a gain factor that can be a matrix or a scalar, $e(n)$ is an estimation error, and $\hat{\phi}(n)$ is the regressor or information vector associated to the respective adaptive algorithm.

The implementation of the above updating equation using the coefficients $\theta_{2\ell}$ of the lattice realization as $\hat{\theta}(n)$ is suggested. The estimation error $e(n)$ and the regressor vector $\hat{\phi}(n)$ are defined as in the direct-form version, but are evaluated using a recursive approach like the one leading to equation (8) which is based on the recursive-in-order equations of lemma 1. Thus, it is guaranteed that $e_\ell(n) = e_d(n)$ and $\hat{\phi}_\ell(n) = \hat{\phi}_d(n)$, and then an entirely equivalent updating process is obtained with the additional feature of enabling pole-monitoring during the convergence process to avoid instability of the adaptive filter. In [7] it was shown that this approach generates a lattice-based algorithm fully consistent with the standard direct-form algorithm in the sense that both methods yield a set of stationary points corresponding to the same input/output description for the adaptive filter. This follows from the fact that the stationary points of an adaptive algorithm are the solutions of the equation:

$$E\left[e(n)\hat{\phi}(n)\right] = 0 \quad \text{(10)}$$

For equivalent direct-form and two-multiplier lattice realizations, the residual errors are automatically equal. Moreover, from the above proposed simplification, the corresponding regression vectors also become identical, and consequently the following result applies:

$$E\left[e_d(n)\hat{\phi}_d(n)\right] = 0 \Longleftrightarrow E\left[e_{2\ell}(n)\hat{\phi}_{2\ell}(n)\right] = 0 \quad \text{(11)}$$

The equivalence between direct-form and simplified lattice algorithms has also been verified through several computer simulations [6], [7] and via analytical methods in [8].

Using the proposed simplification $\hat{\phi}_\ell(n) = \hat{\phi}_d(n)$ for the regressor vector and utilizing the recursive equations (5)-(6), lattice-based algorithms are efficiently implemented, requiring $O(N)$ multiplication/division operations as opposed to the $O(N^2)$ operations required by earlier lattice adaptive algorithms [4], [5]. Thus, the algorithms proposed here, along with the ones presented in

[6]-[8], present similar computational complexity to their equivalent direct-form counterparts with the additional feature of allowing pole-monitoring to be implemented in real time.

Remark: 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 those two structures. Indeed, the normalized lattice structure has the set of coefficients $\boldsymbol{\theta}_{4\ell} = \left[\phi_1 \ \ldots \ \phi_N \ \ h_0^{'} \ \ldots \ h_N^{'}\right]^T$ that are related to the entries of $\boldsymbol{\theta}_{2\ell}$ by the following equations [2]:

$$sin\phi_i = k_i; \ i = 1, \ldots, N \qquad (12a)$$

$$h_j^{'} = \frac{h_j}{\pi_j}; \ j = 0, \ldots, N \qquad (12b)$$

where the parameters π_j are given by:

$$\pi_{j-1} = \pi_j cos\phi_{j-1}; \ j = N-1, \ldots, 1 \qquad (13)$$

with $\pi_N = 1$. Applying those relationships to the equations given in lemma 1, we obtain recursive-in-order equations for the normalized lattice realization equivalent to (5)-(6).

IV. SIMULATIONS

To illustrate the application and generality of the previously proposed methods, the implementation of the *composite square error* (CSE) [9] algorithm is performed here. The CSE algorithm combines the *equation error* (EE) and *output error* (OE) methods [10] in an attempt to obtain the stable convergence and unique solution associated with the EE algorithm and the unbiased global solution characteristic to the OE algorithm. The CSE algorithm is generally described by [9]:

$$\Delta\hat{\boldsymbol{\theta}}(n) = \mu(\gamma e_{EE}(n)\hat{\boldsymbol{\phi}}_{EE}(n) + (1-\gamma)e_{OE}(n)\hat{\boldsymbol{\phi}}_{OE}(n)) \qquad (14)$$

where $\Delta\hat{\boldsymbol{\theta}}(n) = \hat{\boldsymbol{\theta}}(n+1) - \hat{\boldsymbol{\theta}}(n)$ is the increment in the adaptive filter coefficient vector from the instant of time n to $n+1$, and $e(n)$ and $\hat{\boldsymbol{\phi}}(n)$ are respectively the residual error and the information vector associated to the corresponding scheme indicated by their subindeces. Also, γ is the weighting parameter that adjusts the combination scheme between the EE and OE algorithms.

In the simulations performed here, an adaptive two-multiplier lattice filter is used such that $\hat{\boldsymbol{\theta}}_{2\ell}(n) = \left[\hat{k}_1(n) \ \ldots \ \hat{k}_N(n) \ \hat{h}_0(n) \ \ldots \ \hat{h}_N(n)\right]^T$. Moreover, as previously indicated, the error signals and the regressor vectors for the OE and EE schemes are given as standardly defined for the direct-form realization [10], but they are calculated here using the lattice structure coefficients and the recursive-in-order equations given in lemma 1.

Consider then the system identification example described in [4] where the plant is defined as:

$$H(q) = \frac{0.0154+0.0462q^{-1}+0.0462q^{-2}+0.0154q^{-3}}{1-1.99q^{-1}+1.572q^{-2}-0.4583q^{-3}} \qquad (15)$$

which yields the two-multiplier lattice coefficient vector given by:

$$\boldsymbol{\theta}_{2\ell}^T = [-.8756 \ .8355 \ -.4583 \ .0857 \ .1455 \ .0769 \ .0154] \qquad (16)$$

Consider an adaptive filter with $N = 3$ corresponding to a strictly sufficient order identification case and let the input signal be a white noise with zero mean and unity variance. Assume also a perturbation signal statistically independent to the input signal consisting of white noise with zero mean and variance $\sigma_v^2 = 0.007$.

Fig. 1 and Fig. 2 show the convergence trajectories followed by the adaptive coefficients in equation (14) when $\gamma = 1$. This corresponds to the case when the CSE algorithm degenerates into the EE algorithm. It can be observed that due to the presence of a perturbation signal, the solution achieved by these methods was biased with regards to the optimal one represented in these figures by the dotted lines. In this simulation, the value of the convergence parameter was optimized for a faster stable convergence via trial and error and it was equal $\mu_\ell = 0.06$.

When the value of the composite factor is decreased from $\gamma = 1$ to $\gamma = 0$, the coefficient vector bias observed in the previous figures should be eliminated, as the CSE algorithms behaves more and more alike the OE algorithm as γ approaches zero. This fact can be clearly observed in Fig. 3 and Fig. 4 obtained for the case where the value of γ is decreased 0.1 units every 5000 iterations, starting from $\gamma = 1$ until it reaches $\gamma = 0$, remaining null thereafter. For this case, due to the worst stability properties associated to the OE algorithm, the value of the convergence factor had to be reduced to $\mu = 0.01$ to avoid instability during the adaptation process, slowing down the overall convergence speed, but allowing the algorithm to converge to the global optimal solution.

The same simulation example was performed using the direct-form structure to realize the adaptive IIR filter. This direct-form version presented convergence problems due to instability of the adaptive filter during the adaptation process. To eliminate this problem the convergence parameter μ had to be significantly reduced largely increasing the number of iterations to achieve a satisfactory steady-state. This fact clearly demonstrates the improvements that result from implementing adaptive IIR algorithms using the lattice structure.

V. CONCLUSION

In this paper, lattice-based versions of adaptive IIR algorithms were discussed. Relationships of the parameters of the direct-form and two lattice realizations were mentioned. It was shown that those equations lead to the implementation of efficient and consistent lattice-based adaptive IIR algorithms, including members of the equation error family of algorithms. Computer simulations of a composite adaptive IIR algorithm were performed to demonstrate the application and generality of the proposed methods. The results presented here and in others found in the literature indicate that the lattice structure constitutes an important and efficient tool for the realization of real-time adaptive IIR filters.

REFERENCES

[1] A. H. Gray, Jr., and J. D. Markel, "Digital lattice and ladder filter synthesis," *IEEE Trans. Audio Electroacoust.*, vol. AU–21, no 6, pp. 492–500, Dec. 1973.
[2] A. H. Gray, Jr., and J. D. Markel, "A normalized digital filter structure," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP–23, no. 3, pp. 268–277, June 1975.
[3] M. Nayeri, "Lattice form IIR adaptive filters: A structural issue in filtering," *Proc. 32nd. Midwest Symp. Circuits Syst.*, Champaign, USA, pp. 1147–1149, Aug. 1990.
[4] D. Parikh, N. Ahmed, and S. D. Stearns, "An adaptive lattice algorithm for recursive filters," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-28, no. 1, pp. 110–112, Jan. 1980.
[5] S. Horvath, Jr., "Lattice form adaptive recursive digital filters: Algorithms and applications," *Proc. IEEE Int. Symp. Circuits Syst.*, Houston, USA, pp. 128–133, Apr. 1980.
[6] J. A. Rodríguez-Fonollosa and E. Masgrau, "Simplified gradient calculation in adaptive IIR lattice filters," *IEEE Trans. Signal Processing*, vol. 39, no. 7, pp. 1702–1705, July 1991.
[7] P. A. Regalia, "Stable and efficient lattice algorithms for adaptive IIR filtering," *IEEE Trans. Signal Processing*, vol. 40, no. 2, pp. 375–388, Feb. 1992.
[8] K. X. Miao, H. Fan, and M. Doroslovački, "Cascade lattice IIR adaptive filters," *IEEE Trans. Signal Processing*, vol. 42, no. 4, pp. 721–742, Apr. 1994.
[9] S. L. Netto and P. Agathoklis, "A new composite adaptive IIR algorithm," *Proc. 28th Asilomar Conf. Signals, Syst., and Computers*, Pacific Grove, USA, pp. 1506–1510, Oct./Nov. 1994.
[10] J. J. Shynk, "Adaptive IIR filtering," *IEEE Acoust., Speech, Signal Processing Magazine*, vol. 6, no. 2, pp. 4–21, Apr. 1989.

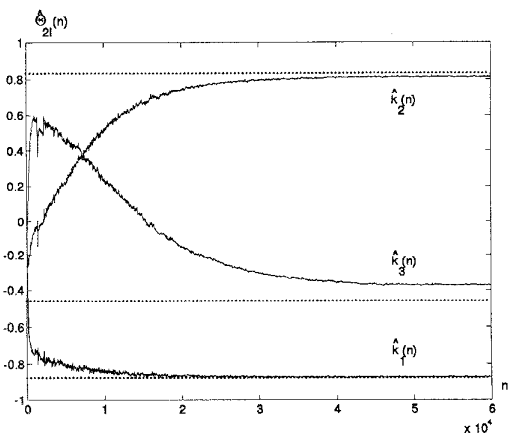

Fig. 1: CSE algorithm - $\gamma = 1$ - $\hat{k}(n)$ coefficients.

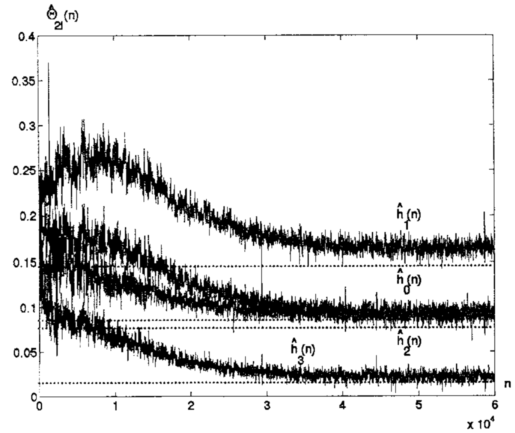

Fig. 2: CSE algorithm - $\gamma = 1$ - $\hat{h}(n)$ coefficients.

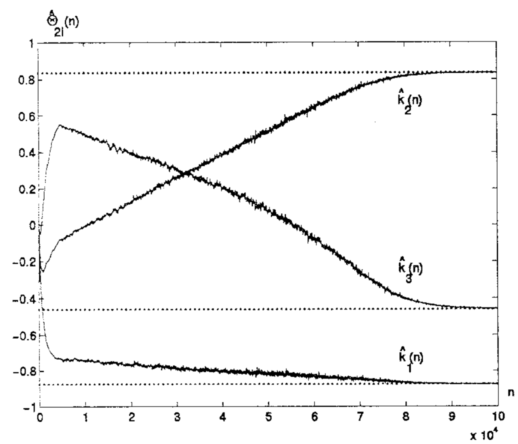

Fig. 3: CSE algorithm - Decreasing γ - $\hat{k}(n)$ coefficients.

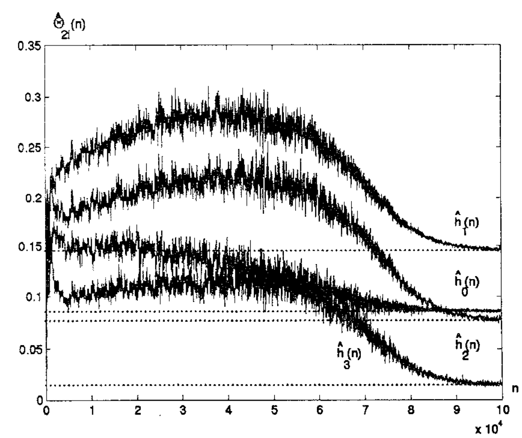

Fig. 4: CSE algorithm - Decreasing γ - $\hat{h}(n)$ coefficients.