Automatica, Vol. 27, No. 5, pp. 783–798, 1991
Printed in Great Britain.

0005–1098/91 $3.00 + 0.00
Pergamon Press plc

Locally Robust Identification of Linear Systems Containing Unknown Gain Elements with Application to Adapted IIR Lattice Models*

GEOFFREY A. WILLIAMSON,† C. RICHARD JOHNSON, JR‡
and BRIAN D. O. ANDERSON§

A robust, local convergence property, established for a recursive algorithm identifying unknown gain parameters in a linear system, is applied to adapted IIR lattice models having desirable properties related to stability maintenance and numerical insensitivity.

Key Words—Adaptive systems; system identification; lattice filters; parameter estimation; stability.

Abstract—We consider the identification of stable linear systems whose unknown parameters may be interpreted as feedback gains. By using the output error between the true system and a model of it containing adjustable parameters, we develop a recursive algorithm for estimating the unknown parameters. We describe a persistency of excitation condition on the system input which guarantees a robust, local convergence property for the algorithm. We then apply our results to the identification of the parameters of a tapped lattice model of a linear, infinite-impulse response (IIR) plant. Considering the identification of a lattice, rather than a direct form, model of a linear system is attractive due to (i) the simplicity of its crucial stability check and maintenance procedure for the adapted IIR parametrization and (ii) the numerical insensitivity properties of the lattice structure. Reproducible simulation evidence is presented that supports our results.

1. INTRODUCTION

IN THE first half of this paper, we consider the recursive identification of linear systems represented by a model class in which the unknown parameters appear as (internal) feedback gains. This model class [which we label the Feedback Gain Model (FGM)] is quite general, and includes multi-input multi-output in addition to single-input single-output systems. One may recast typical recursive identification schemes, aimed at identifying the coefficients of a finite impulse response (FIR) model or the coefficients of the numerator and denominator of a transfer function representation, in terms of the FGM. Using the FGM, one may also phrase the identification of the parameters in a variety of realization structures (direct form, lattice, parallel, cascade) for single-input single-output transfer functions. An important additional feature of the FGM is that it facilitates the development of identification schemes for linear systems in which some, but not all, of the system parameters are known in advance.

To be represented by a FGM, a parametrized linear system must possess the following crucial property. For each unknown parameter α, there must exist transfer function matrices $G_{ij}(z)$ such that the overall transfer function from the vector input u to the vector output y is

$$W(z) = G_{11}(z) - \alpha G_{12}(z)[I + \alpha G_{22}(z)]^{-1} G_{21}(z). \tag{1.1}$$

(See Fig. 1.) Moreover, the $G_{ij}(z)$ are independent of α (but may depend on some other parameters). Of course, the FGM may deal with more than one unknown parameter; we will generalize (1.1) to cope with this situation in Section 2.

We develop a candidate recursive algorithm for estimating the unknown parameters in a FGM. This algorithm is based on the output

* Received 25 October 1988; revised 2 May 1990; received in final form 1 December 1990. This work is based on two papers presented at the 8th IFAC Symposium on Identification and System Parameter Estimation, Beijing, China, 1988: C. R. Johnson, Jr and B. D. O. Anderson, "Identification of a tapped lattice representation of a linear system" and B. D. O. Anderson and C. R. Johnson, Jr, "Identification of linear systems containing unknown linear gain elements." The published proceedings of this IFAC Meeting may be ordered from Pergamon Press plc, Headington Hill Hall, Oxford OX3 0BW, U.K. This paper was recommended for publication in revised form by Associate Editor R. Lozano-Leal under the direction of Editor P. C. Parks.

† Department of Electrical and Computer Engineering, Illinois Institute of Technology, Chicago, IL 60616, U.S.A. Author to whom all correspondence should be addressed.

‡ School of Electrical Engineering, Phillips Hall, Cornell University, Ithaca, NY 14853, U.S.A.; supported by NSF grants MIP-8608787 and MIP-8921003.

§ Department of Systems Engineering, Australian National University, GPO Box 4, Canberra, ACT 2601, Australia.

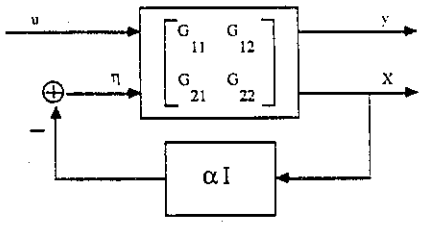

FIG. 1. System input and output are u and y, and α denotes an unknown parameter.

error formulation (Landau, 1976) of the identification problem, and it may be interpreted as an approximate gradient descent of the average squared output error surface. We relate the structure of a (fictional) system which generates the output error to that of the FGM for the unknown system, and we then exploit this structural feature of the FGM in the generation of sensitivity functions to be used in the algorithm. For this algorithm, we establish local exponential convergence of the parameter estimates, via techniques similar to those in Anderson *et al.* (1986) and Sethares *et al.* (1989), given satisfaction of a persistency of excitation condition. The exponential nature of the convergence then imparts a degree of robustness of the parameter convergence to the effects of output measurement noise, undermodeling, etc.

The general formulation of this type of adaptive algorithm, together with its local stability analysis, has been discussed in, for example, Anderson *et al.* (1986), Ljung and Söderström (1983), and Narendra *et al.* (1985). Also, the identification of partially known systems in certain special situations is considered in Dasgupta *et al.* (1984, 1986). A general state-space formulation of the identification problem, which may encompass FGM systems, appears in Ljung and Söderström (1983). There, however, one does not isolate the unknown portion of the system model. Furthermore, the specific structural aspects of the FGM and its sensitivity function generation are not exploited. In this current work, we fully detail the adaptive algorithm specification, together with the local convergence proof, for the general FGM. We draw connections which link the FGM structure to the generation of sensitivity functions implicit in the algorithm specification, and we are thus able to provide a complete formulation of a locally stable adaptive algorithm.

The second half of the paper applies the results for the general FGM to the identification of the parameters of a tapped lattice model of a linear, infinite impulse response (IIR) plant. The lattice model identification serves as an example of the FGM identification, but it is also of independent interest. The tapped lattice model (Gray and Markel, 1973) is an attractive alternative to the direct form model in the identification of a linear system because of (i) the simplicity of its crucial stability check and maintenance procedure for the adapted IIR parametrization, and (ii) the numerical insensitivity properties of the lattice structure.

These advantages make the lattice model useful in the adaptive IIR filter problem setting (Johnson, 1984). In fact, a gradient descent-based algorithm for adaptive IIR lattice filters has already been developed (Parikh *et al.*, 1980). Missing, however, is an examination of the convergence properties of this algorithm that dovetails nicely with convergence results for the direct form structure. Here, we show that the algorithm for the lattice model obtained using the FGM approach is identical with the algorithm of Parikh *et al.* (1980). Therefore, we may invoke the stability analysis for the FGM identification scheme in order to provide the first convergence theory for the algorithm of Parikh *et al.* (1980). A further consequence of this use of the FGM identification theory is the justification of the seemingly *ad hoc* approximations required in the sensitivity function calculations leading to the algorithm of Parikh *et al.* (1980).

A note of caution: one should recognize that we are not working here with the adaptive lattice as a whitening filter. In such a case, one typically uses the adaptive lattice to filter the output of an autoregressive model driven by white noise, as in Ljung and Söderström (1983), with the lattice representing a moving average (FIR) filtering operation. This usage of the adaptive lattice as a whitening filter is, of course, common (Friedlander, 1982). Here, however, we assume an I/O model where we have access to both input and output, and the lattice filter has an IIR form.

The paper is organized as follows. In Section 2, we develop the candidate adaptive algorithm for the recursive identification of the unknown parameters in a FGM. Our examination includes a discussion of the output error construction, the sensitivities of the output error with respect to the parameter estimates, and finally the development of the candidate recursive algorithm. In Section 3, we present the convergence analysis of the algorithm, demonstrating its local exponential stability under persistent excitation.

We apply the results for the general FGM identification problem to the adaptive IIR lattice filter in Section 4. We show that the algorithm development indicated in Section 2 leads, for the adaptive IIR lattice, to the same basic algorithm of Parikh *et al.* (1980). Thus, the convergence results of Section 3 then provide the algorithm of

Parikh *et al.* (1980) with its first formal convergence analysis. Section 5 presents simulation evidence demonstrating the local stability and robustness of the algorithm. Section 6 provides some closing remarks.

2. RECURSIVE IDENTIFICATION OF A FEEDBACK GAIN MODEL FOR A LINEAR SYSTEM

As noted in the introduction, a Feedback Gain Model (FGM) may be used to represent a parametrized class of linear systems if for each parameter α, the system transfer function may be written as

$$W(z) = G_{11}(z) - \alpha G_{12}(z)[I + \alpha G_{22}(z)]^{-1} G_{21}(z); \tag{2.1}$$

i.e. Fig. 1 depicts the system. We have deliberately cast our discussion in discrete time here, since the IIR lattice application of interest here is itself in discrete time. However, we believe the problem origin may in many occasions necessitate a continuous time setting. Often, it is the case that physical parameters (such as mass, friction coefficient, mutual inductance, moment of inertia, reaction constant and so on) can appear quite simply in a continuous time model, but get "smeared out" in the associated discrete-time model. For a continuous time formulation, refer to Anderson and Johnson (1988).

When the feedback loop in Fig. 1 involves scalar X and η, we have a situation termed rank-1 dependence in Dasgupta *et al.* (1984, 1986), and when such dependence applies for all parameters, special procedures for identification are available, as in Dasgupta *et al.* (1984, 1986). In a continuous-time description of a physical system, one can associate rank-1 dependence with a mass or friction coefficient, but not with, for example, mutual inductance. (Actually, sometimes the inverse of the physical quantity must be used, rather than the quantity itself, to secure the rank-1 property, but this is a minor point.) In the IIR lattice application considered later in this paper, the lattice coefficients each involve a rank-2 dependence, and the taps a rank-1 dependence. In this section, we make no assumption on the order of the rank dependence.

We generalize the system in Fig. 1 to include all the unknown parameters, as shown in Fig. 2. In Fig. 2, the matrix A is given by

$$A = \text{diag}\,\{\alpha_1 I_{n_1}, \alpha_2 I_{n_2}, \ldots, \alpha_m I_{n_m}\}. \tag{2.2}$$

Note that, if $n_1 = n_2 = \cdots = n_m = 1$, then each α_i has a rank-1 dependence. The transfer function

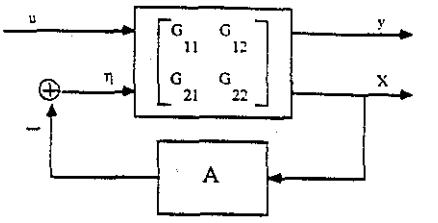

FIG. 2. System input and output are u and y, and A contains a number of unknown parameters.

matrix from input to output is

$$W(z) = G_{11}(z) - G_{12}(z)A(I + G_{22}(z)A)^{-1}G_{21}(z) \tag{2.3}$$

or

$$W(z) = G_{11}(z) - G_{12}(z)(I + AG_{22}(z))^{-1}AG_{21}(z), \tag{2.4}$$

and is assumed stable without further explicit comment. The driving signal $u(\cdot)$ is assumed bounded, as are all internal signals, including $y(\cdot)$, $X(\cdot)$ and $\eta(\cdot)$. The transfer function matrices $G_{ij}(z)$ are assumed known, while the constants $\alpha_1, \ldots, \alpha_m$ appearing in A are the unknowns which are to be identified.

A great many adaptive identification procedures work by copying the structure of the plant into a model which uses an adjustable known parameter at every point where the plant has a fixed unknown parameter. This strategy is followed here. Figure 3 depicts the adjustable model, in which

$$\hat{A}(k) = \text{diag}\,\{\hat{\alpha}_1(k) I_{n_1}, \hat{\alpha}_2(k) I_{n_2}, \ldots, \hat{\alpha}_m(k) I_{n_m}\}. \tag{2.5}$$

The adjustable model is driven by the same signal $u(\cdot)$ as the true plant. Notice that one or more of the entries of $\hat{X}$ and X may coincide; in equation error identification of an entire transfer function, $\hat{X} \equiv X$ (the entries being delayed version of u and y). In output error identification, some, but not all, of the entries are identical (those corresponding to delayed versions of u).

A vital quantity of concern in any adaptive algorithm is the output error, here $y - \hat{y}$. Figure 4 sums up an important result showing how this depends on the parameter error.

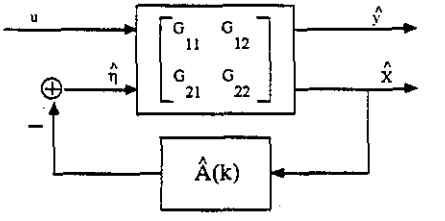

FIG. 3. The adjustable model.

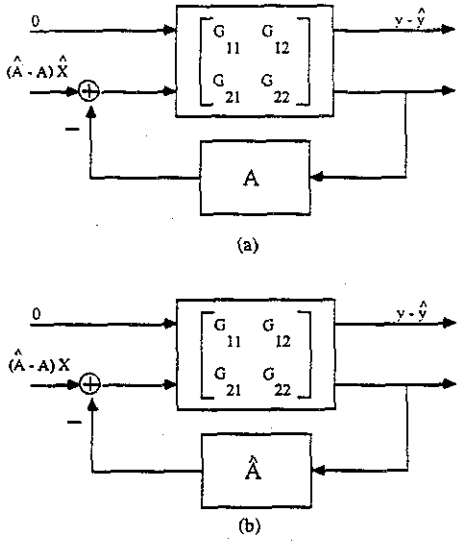

FIG. 4. Two representations of the large scale sensitivity.

Theorem 1. With plant as depicted in Fig. 2 and adjustable model as in Fig. 3, suppose that the adjustable model is stable with $\hat{A}(k)$ fixed at $\hat{A}$. Then (modulo decaying initial condition effects) each of the two arrangements depicted in Fig. 4 generates the error $y-\hat{y}$.

Proof. Using (2.4) in $y = Wu$ and an expresssion similar to (2.3) in $\hat{y} = \hat{W}u$,

$$\begin{aligned} y-\hat{y} &= G_{12}\hat{A}(I+G_{22}\hat{A})^{-1}G_{21}u \\ &\quad - G_{12}(I+AG_{22})^{-1}AG_{21}u \\ &= G_{12}\{\hat{A}(I+G_{22}\hat{A})^{-1} - (I+AG_{22})^{-1}A\}G_{21}u \\ &= G_{12}(I+AG_{22})^{-1}\{(I+AG_{22})\hat{A} \\ &\quad - A(I+G_{22}A)\}(I+G_{22}\hat{A})^{-1}G_{21}u \\ &= G_{12}(I+AG_{22})^{-1}(\hat{A}-A)(I+G_{22}\hat{A})^{-1}G_{21}u. \end{aligned} \tag{2.6}$$

Similarly,

$$y-\hat{y} = G_{12}(I+\hat{A}G_{22})^{-1}(\hat{A}-A)(I+G_{22}A)^{-1}G_{12}u. \tag{2.7}$$

Now for Fig. 3,

$$\hat{X} = (I+G_{22}\hat{A})^{-1}G_{21}u \tag{2.8}$$

while for Fig. 2,

$$X = (I+G_{22}A)^{-1}G_{21}u. \tag{2.9}$$

With inputs of zero and $(\hat{A}-A)\hat{X}$ used as in Fig. 4(a), one can establish that the upper output is given by the right side of (2.6). Similarly, with inputs of zero and $(\hat{A}-A)X$ used as in Fig. 4(b), one can establish that the upper output is given by the right side of (2.7). Thus the theorem claim is established. □

Our goal now is to exploit the error formulae of (2.6)–(2.9) to generate an algorithm for recursively identifying the unknown coefficients α_i from (2.2)–(2.4). We showed that

$$\begin{aligned} y-\hat{y} &= G_{12}(I+AG_{22})^{-1}(\hat{A}-A)\hat{X} \\ &= G_{12}(I+AG_{22})^{-1}\,\mathrm{diag}\,\{\hat{X}_i\} \begin{bmatrix} \hat{\alpha}_1-\alpha_1 \\ \vdots \\ \hat{\alpha}_1-\alpha_1 \\ \vdots \\ \hat{\alpha}_m-\alpha_m \\ \vdots \\ \hat{\alpha}_m-\alpha_m \end{bmatrix}. \end{aligned} \tag{2.10}$$

Note that the term $\hat{\alpha}_i-\alpha_i$ occurs in n_i consecutive entries for $i=1,\ldots,m$. Suppose for simplicity that y is scalar. Then

$$\begin{aligned} &y(k)-\hat{y}(k) \\ &= [\chi_1^1(k)\cdots\chi_1^{n_1}(k)\cdots\cdots\chi_m^1(k)\cdots\chi_m^{n_m}(k)] \\ &\quad \times \begin{bmatrix} \hat{\alpha}_1-\alpha_1 \\ \vdots \\ \hat{\alpha}_1-\alpha_1 \\ \vdots \\ \hat{\alpha}_m-\alpha_m \\ \vdots \\ \hat{\alpha}_m-\alpha_m \end{bmatrix} \\ &= [\psi_1(k)\cdots\psi_m(k)] \begin{bmatrix} \hat{\alpha}_1-\alpha_1 \\ \vdots \\ \hat{\alpha}_m-\alpha_m \end{bmatrix} \end{aligned} \tag{2.11}$$

where

$$\psi_i(k) = \sum_{j=1}^{n_i} \chi_i^j(k). \tag{2.12}$$

Notice that the signals $\chi_i^j(k)$ and then the signals $\psi_i(k)$ can be thought of as arising through excitation of the true system (at other than a standard input point) with signals measured in the identifier, or adjustable model. More

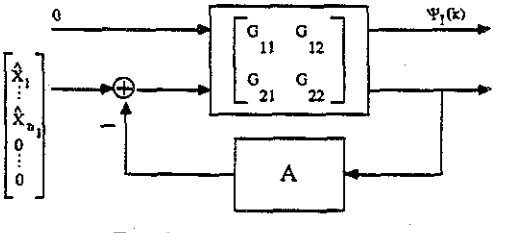

FIG. 5. Generation of $\psi_1(k)$.

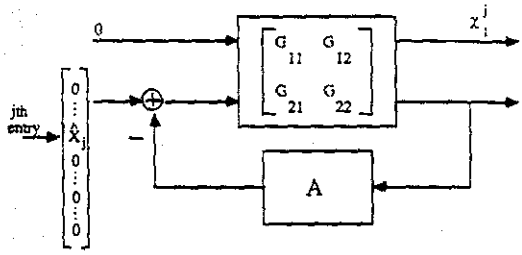

FIG. 6. Generation of $\chi_1^j(k)$.

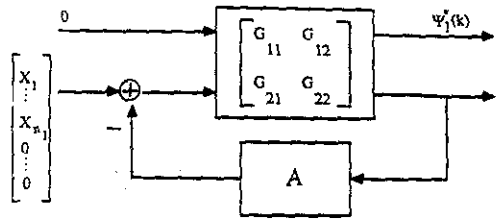

FIG. 8. Generation of $\psi_1^*(k)$.

precisely, Figs 5 and 6 illustrate the calculation of both $\psi_1(k)$ and $\chi_1^j(k)$.

The error $y-\hat{y}$ is precisely constructable using the unknown system output and the adjustable model. Because we do not know the parameters α_i, the individual signals $\psi_i(k)$ cannot be computed. We can however generate approximations $\hat{\psi}_i(k)$ of the $\psi_i(k)$ by using $\hat{\alpha}_i$ in place of α_i in the generating process. Figure 7 shows the idea, and it should be compared with Fig. 5.

Now were the ψ_i available, a reasonable choice for an update algorithm would be

$$\begin{bmatrix} \hat{\alpha}_1(k+1) \\ \vdots \\ \hat{\alpha}_m(k+1) \end{bmatrix} = \begin{bmatrix} \hat{\alpha}_1(k) \\ \vdots \\ \hat{\alpha}_m(k) \end{bmatrix} - \mu \begin{bmatrix} \psi_1(k) \\ \vdots \\ \psi_m(k) \end{bmatrix} [y(k)-\hat{y}(k)] \quad (2.13)$$

because this would imply

$$\begin{bmatrix} \hat{\alpha}_1(k+1) \\ \vdots \\ \hat{\alpha}_m(k+1) \end{bmatrix} = \begin{bmatrix} \hat{\alpha}_1(k) \\ \vdots \\ \hat{\alpha}_m(k) \end{bmatrix} - \mu \begin{bmatrix} \psi_1(k) \\ \vdots \\ \psi_m(k) \end{bmatrix} \times [\psi_1(k) \cdots \psi_m(k)] \begin{bmatrix} \hat{\alpha}_1(k)-\alpha_1 \\ \vdots \\ \hat{\alpha}_m(k)-\alpha_m \end{bmatrix}, \quad (2.14)$$

which is well studied (Anderson *et al.*, 1986; Weiss and Mitra, 1979; Narendra *et al.*, 1985). As just noted though, $\psi_i(k)$ is not available. So

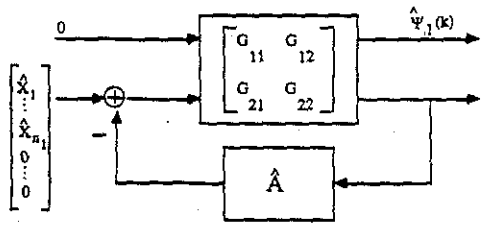

FIG. 7. Generation of $\hat{\psi}_1(k)$.

we use instead

$$\begin{bmatrix} \hat{\alpha}_1(k+1) \\ \vdots \\ \hat{\alpha}_m(k+1) \end{bmatrix} = \begin{bmatrix} \hat{\alpha}_1(k) \\ \vdots \\ \hat{\alpha}_m(k) \end{bmatrix} - \mu \begin{bmatrix} \hat{\psi}_1(k) \\ \vdots \\ \hat{\psi}_m(k) \end{bmatrix} [y(k)-\hat{y}(k)]. \quad (2.15)$$

Of course, μ is a positive constant, and can be replaced by diag $\{\mu_1, \mu_2, \ldots, \mu_m\}$ with $\mu_i > 0$. Note that the signals $\hat{\psi}_i(k)$ have been defined as if the $\hat{\alpha}_i$ were constant, and now we have allowed for time variation of the $\hat{\alpha}_i$. Obviously the definition of $\hat{\psi}_i(k)$ is immediately extendable to this situation. But also, if μ is small enough so that the $\hat{\alpha}_i$ vary sufficiently slowly, the use of frequency-domain notation, as in, for example, (2.8) and (2.23) below (carrying with it the implication of constant $\hat{\alpha}$), can be justified (Anderson *et al.*, 1986).

Above we have defined signals $\psi_i(k)$ and $\hat{\psi}_i(k)$. For completeness, we define a third signal $\psi_i^*(k)$. Figure 8 illustrates the idea:

$$\psi_1^*(k) = G_{12}(I + AG_{22})^{-1}[X_1 \cdots X_{n_1} \quad 0 \cdots 0]^T. \quad (2.16)$$

We note that the quantities $\hat{\psi}_i(k)$ and $\psi_i^*(k)$ have interpretations as sensitivity coefficients:

$$\frac{\partial y(k)}{\partial \alpha_i} = -\psi_i^*(k) \quad (2.17)$$

$$\frac{\partial \hat{y}(k)}{\partial \hat{\alpha}_i} = -\hat{\psi}_i(k). \quad (2.18)$$

For example, for (2.17) in the scalar case (with reference to Fig. 1) we have

$$\frac{\partial}{\partial \alpha} y = \frac{\partial}{\partial \alpha}[G_{11}u + G_{12}\eta] = G_{12}\frac{\partial}{\partial \alpha}\eta, \quad (2.19)$$

$$\frac{\partial}{\partial\alpha}\eta = \frac{\partial}{\partial\alpha}[-\alpha X]$$
$$= -X - \alpha\frac{\partial}{\partial\alpha}X$$
$$= -X - \alpha\frac{\partial}{\partial\alpha}[G_{21}u + G_{22}\eta]$$
$$= -[I + \alpha G_{22}]^{-1}X, \qquad (2.20)$$

and hence combining (2.19) and (2.20) yields

$$\frac{\partial}{\partial\alpha}y = -G_{12}[I + \alpha G_{22}]^{-1}X$$
$$= -\psi^*, \qquad (2.21)$$

where ψ^* relates to X as in Fig. 8.

Medanic and Kokotovic (1965) and Kokotovic *et al.* (1966) discuss the calculation of sensitivities like (2.17) and (2.18) in certain special cases via procedures similar to those of this paper, and indicate their relevance for adaptive control. Note that (2.18) enables interpretation of the algorithm (2.15) as

$$\hat{\alpha}_i(k+1) = \hat{\alpha}_i(k) - \mu\frac{\partial}{\partial\hat{\alpha}_i(k)}[\tfrac{1}{2}[y(k) - \hat{y}(k)]^2], \qquad (2.22)$$

which is a gradient descent algorithm. Since (2.18) is calculated assuming that $\hat{\alpha}_i$ is constant, (2.15) is actually only an approximation of (2.22).

For completeness, we also note an alternative arrangement for generating $\psi_i(k)$. As an alternative to (2.10) and in analogy with the development of (2.10), we have

$$y - \hat{y} = G_{12}(I + \hat{A}G_{22})^{-1}\,\mathrm{diag}\,\{X_i\}\begin{bmatrix}\hat{\alpha}_1 - \alpha_1\\ \vdots\\ \hat{\alpha}_1 - \alpha_1\\ \vdots\\ \hat{\alpha}_m - \alpha_m\\ \vdots\\ \hat{\alpha}_m - \alpha_m\end{bmatrix}. \qquad (2.23)$$

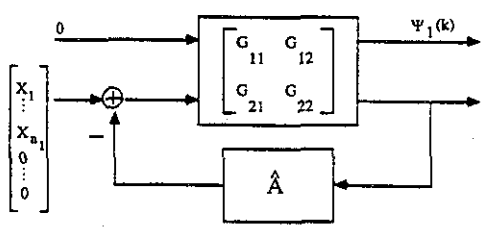

FIG. 9. Alternate arrangement for generating $\psi_1(k)$.

Now (2.11) also remains valid. It follows then that $\psi_i(k)$ could be generated by replacing A and $\hat{X}_i$ in Fig. 5 by $\hat{A}$ and X_i (see Fig. 9).

3. LOCAL CONVERGENCE OF THE ADAPTIVE ALGORITHM

Here we establish the local stability of the adaptive algorithm of Section 2 by examining the corresponding error system. The basic idea is that local stability for the error system (and hence the algorithm) will follow from the exponential stability of its linearized version. We will find that the linearized error system has the form of

$$x(k+1) = [I - \mu\phi(k)\phi^T(k)]x(k). \qquad (3.1)$$

Equation (3.1) is exponentially stable given a persistently exciting $\phi(k)$, when $\phi(k)$ does not depend on x (Anderson *et al.*, 1986; Johnson, 1988). This fact, applied to our linearized error system, gives a condition for local stability of the error system coresponding to the algorithm of (2.15).

To obtain the error system corresponding to the algorithm (2.15), subtract both sides of (2.15) from a vector of the true parameters α_i. With $\tilde{\alpha}_i(k) = \hat{\alpha}_i(k) - \alpha_i$, we have

$$\begin{bmatrix}\tilde{\alpha}_1(k+1)\\ \vdots\\ \tilde{\alpha}_m(k+1)\end{bmatrix} = \begin{bmatrix}\tilde{\alpha}_1(k)\\ \vdots\\ \tilde{\alpha}_m(k)\end{bmatrix} + \mu\begin{bmatrix}\hat{\psi}_1(k)\\ \vdots\\ \hat{\psi}_m(k)\end{bmatrix}[y(k) - \hat{y}(k)]$$
$$\approx \left[I - \mu\begin{bmatrix}\hat{\psi}_1(k)\\ \vdots\\ \hat{\psi}_m(k)\end{bmatrix}\times[\psi_1(k)\cdots\psi_m(k)]\right]\begin{bmatrix}\tilde{\alpha}_1(k)\\ \vdots\\ \tilde{\alpha}_m(k)\end{bmatrix}. \qquad (3.2)$$

There is only approximate equality in the second equation of (3.2) in order to reflect the assumptions of fixed $\hat{\alpha}_i$ in (2.11). For slow adaptation (small μ), this approximation is a good one: with a small enough step size, the difference of (3.2) from the true error equation will be $O(\mu^2)$.

Now, consider the related error equation

given by

$$\begin{bmatrix} \tilde{\alpha}_1(k+1) \\ \vdots \\ \tilde{\alpha}_m(k+1) \end{bmatrix} = \begin{bmatrix} I - \mu \begin{bmatrix} \psi_1^*(k) \\ \vdots \\ \psi_m^*(k) \end{bmatrix} \end{bmatrix} \times [\psi_1^*(k) \cdots \psi_m^*(k)] \begin{bmatrix} \tilde{\alpha}_1(k) \\ \vdots \\ \tilde{\alpha}_m(k) \end{bmatrix}. \tag{3.3}$$

Observe (see especially Figs 5, 7, 8 and 9) that (with the bold notation denoting a vector quantity)

$$\|\hat{\boldsymbol{\psi}}(k) - \boldsymbol{\psi}^*(k)\| = O\left(\sup_{l \le k} \|\tilde{\boldsymbol{\alpha}}(l)\|\right) \tag{3.4}$$

$$\|\boldsymbol{\psi}(k) - \boldsymbol{\psi}^*(k)\| = O\left(\sup_{l \le k} \|\tilde{\boldsymbol{\alpha}}(l)\|\right) \tag{3.5}$$

when all signals remain bounded. With the system of Fig. 2, and hence that of Fig. 8, presumed stable, boundedness of u guarantees boundedness of $\boldsymbol{\psi}^*(k)$. To assure boundedness of $\hat{\boldsymbol{\psi}}(k)$, one typically constrains the adaptation of the $\hat{\alpha}_i(k)$ parameters so that the adapted model of Fig. 3, and hence that of Fig. 7, has a stable parametrization for each k (Johnson, 1984; Sethares *et al.*, 1989). With these boundedness conditions satisfied, and as $\boldsymbol{\psi}^*$ is independent of $\hat{\boldsymbol{\alpha}}$, it follows from (3.4) and (3.5) that (3.3) is a linearized version of (3.2).

Noting that linearized error system (3.3) has the form of (3.1), we see that (3.3) is exponentially stable under the assumption that $\boldsymbol{\psi}^*$ is persistently exciting. Accordingly, (3.2) also possesses the exponential stability property, but only in a local sense (Weiss and Mitra, 1979; Hahn, 1967; Desoer and Vidyasagar, 1975; Vidyasagar, 1978). Now we are guaranteed a local stability property for our algorithm.

The question of when $\boldsymbol{\psi}^*$ is persistently exciting must be addressed separately. This issue is considered in the continuous-time case in Anderson and Johnson (1988), where persistent excitation follows from (i) the input containing a sufficient number of distinct sinusoids and (ii) dissatisfaction of a type of local unidentifiability condition. This unidentifiability condition is that there exists a direction $\boldsymbol{\gamma}$ (in parameter space) such that the transfer function taking u to $\hat{y}$ is independent of first order variations in the parametrization $\hat{\boldsymbol{\alpha}}$ in the direction $\boldsymbol{\gamma}$, for all input frequencies. Translation of this result to the discrete-time case is straightforward.

We would also like to note that some global convergence results may be obtained if one replaces μ in (2.15) with a suitably decreasing, time-varying step size $\mu(t)$: see, for instance, Chapter 4 of Ljung and Söderström (1983). In such a case, however, the parameters may converge to a local minimum of the corresponding average squared output error surface, or to a point in the parameter space on the boundary between stable and unstable adapted models. Furthermore, choice of a decreasing step size entails a loss of adaptibility which is not suffered with a constant step size.

4. LINEAR SYSTEM IDENTIFICATION WITH AN ADAPTIVE IIR TAPPED LATTICE MODEL

The system identification framework of Sections 2 and 3 may be applied in the identification of the parameters in the "two-multiplier" tapped lattice model of Gray and Markel (1973). This model, with input $u(k)$ and output $y(k)$, is given by

$$y(k) = \sum_{i=0}^{N} v_i B_i(k) \tag{4.1}$$

$$B_i(k) = B_{i-1}(k-1) + w_i F_{i-1}(k), \quad i = 1, \ldots, N \tag{4.2}$$

$$F_i(k) = F_{i+1}(k) - w_{i+1} B_i(k-1), \quad i = 0, \ldots, N-1 \tag{4.3}$$

$$F_N(k) = u(k) \tag{4.4}$$

$$B_0(k) = F_0(k), \tag{4.5}$$

the third-order case of which is depicted in Fig. 10. For this model, stability is assured if and only if the w_i all have magnitude less than one. This feature trivializes the stability maintenance of an adaptive version of (4.1)–(4.5), since one may easily maintain stability by preventing updates of $\hat{w}_i(k)$ to values greater than one in magnitude. (Tests for stability of an adapted direct form model are more complicated. Furthermore, to prevent an update to an unstable parametrization, one must freeze all the denominator parameters, contrasting with the freezing only of individual $\hat{w}_i(k)$ in the lattice model.) Another potential benefit of the lattice form is its reduced sensitivity of pole locations to parameter variation in comparison to the sensitivity of the direct form.

Note also that the parameters v_i and w_i of (4.1)–(4.5) can be related uniquely to the parameters of a direct form model of the same order. See, e.g. Widrow and Stearns (1985). This one-to-one relationship between direct form and tapped lattice model parameters suggests that excitation in $\{u\}$, which is sufficient to solve exactly, given the associated $\{y\}$, for the unknown parameters in a direct form model, will prove sufficient to solve uniquely for the unknown v_i and w_i in (4.1)–(4.5).

To place the lattice model of (4.1)–(4.5) in the

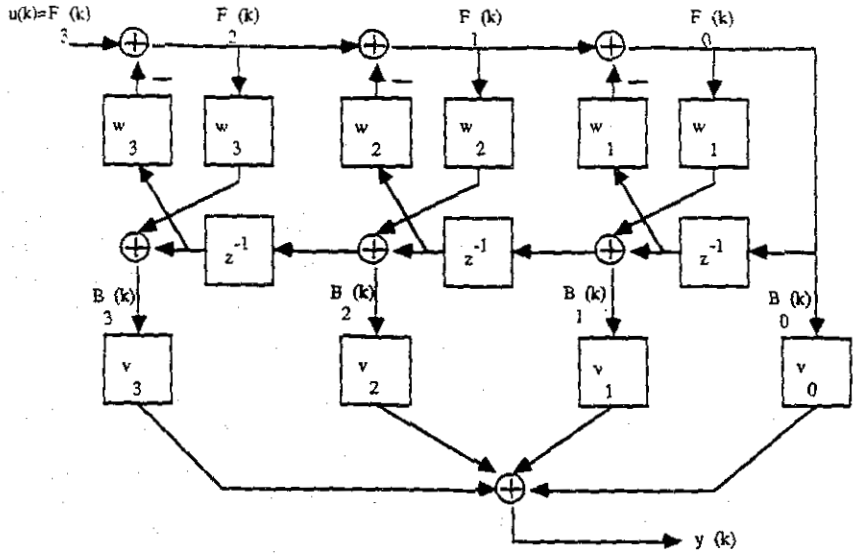

FIG. 10. Third-order tapped IIR lattice.

context of the FGM of Section 2, set

$$A = \operatorname{diag}\{v_0, \ldots, v_n, w_1, \ldots, w_n, w_1, \ldots, w_n\}, \tag{4.6}$$

$$X(k) = [B_0(k) \cdots B_n(k) \quad B_0(k-1) \cdots B_{n-1}(k) \quad F_0(k) \cdots F_{n-1}(k-1)]^T. \tag{4.7}$$

(We have slightly rearranged the ordering of the parameters in (4.6) as compared to (2.2) simply for convenience.) Generalizing from Fig. 10, we have $B_i(k)$ given by

$$\begin{aligned}
B_0(k) &= -w_1 B_0(k-1) - \cdots \\
&\quad - w_n B_{n-1}(k-1) + u(k) \\
B_1(k) &= w_1 F_0(k) + q^{-1}[-w_1 B_0(k-1) - \cdots \\
&\quad - w_n B_{n-1}(k-1)] + q^{-1}u(k) \\
B_2(k) &= w_2 F_1(k) + q^{-1} w_1 F_0(k) \\
&\quad + q^{-2}[-w_1 B_0(k-1) - \cdots \\
&\quad - w_n B_{n-1}(k-1) \\
&\quad + q^{-2}u(k) \\
&\vdots \\
B_n(k) &= w_n F_{n-1}(k) + q^{-1} w_{n-1} F_{n-2}(k) + \cdots \\
&\quad + q^{-(n-1)} w_1 F_0(k) \\
&\quad + q^{-n}[-w_1 B_0(k-1) - \cdots \\
&\quad - w_n B_{n-1}(k-1)] \\
&\quad + q^{-n}u(k),
\end{aligned} \tag{4.8}$$

and $F_i(k)$ given by

$$\begin{aligned}
F_0(k) &= -w_1 B_0(k-1) - \cdots - w_n B_{n-1}(k-1) + u(k) \\
F_1(k) &= -w_2 B_1(k-1) - \cdots - w_n B_{n-1}(k-1) + u(k) \\
&\vdots \\
F_{n-1}(k) &= -w_n B_{n-1}(k-1) + u(k).
\end{aligned} \tag{4.9}$$

From (4.6)–(4.9), we see that with

$$G_{11}(z) = 0 \tag{4.10a}$$

$$G_{12}(z) = [-1 \cdots -1 \quad 0 \cdots \quad 0 \cdots 0] \tag{4.10b}$$

$$G_{21}(z) = [1 \quad z^{-1} \cdots z^{-n} \quad z^{-1} \cdots z^{-n} \quad 1 \cdots 1]^T \tag{4.10c}$$

$$G_{22}(z) = \begin{bmatrix}
 & 1 & 1 & \cdots & 1 & 0 & 0 & \cdots & 0 \\
 & z^{-1} & z^{-1} & \cdots & z^{-1} & -1 & 0 & \cdots & 0 \\
 & z^{-2} & z^{-2} & \cdots & z^{-2} & -z^{-1} & -1 & & \vdots \\
0 & \vdots & \vdots & & \vdots & \vdots & & \ddots & 0 \\
 & z^{-n} & z^{-n} & \cdots & z^{-n} & -z^{-(n-1)} & \cdots & -z^{-1} & -1 \\
 & z^{-1} & z^{-1} & \cdots & z^{-1} & 0 & 0 & \cdots & 0 \\
 & z^{-2} & z^{-2} & \cdots & z^{-2} & -z^{-1} & 0 & \cdots & 0 \\
 & \vdots & \vdots & & \vdots & -z^{-2} & -z^{-1} & & \vdots \\
0 & \vdots & \vdots & & \vdots & \vdots & & \ddots & 0 \\
 & z^{-n} & z^{-n} & \cdots & z^{-n} & -z^{-(n-1)} & \cdots & -z^{-2} & -z^{-1}
\end{bmatrix}, \tag{4.10d}$$

$$\begin{bmatrix}
1 & 1 & \cdots & 1 \\
0 & 1 & \cdots & 1 \\
\vdots & & \ddots & \vdots \\
0 & \cdots & 0 & 1
\end{bmatrix}$$

$$\begin{bmatrix}
0 \\
\vdots \\
0
\end{bmatrix}$$

the lattice model of (4.1)–(4.5) has a FGM representation as in Fig. 2.

To construct the adaptive version of the lattice, we simply replace A and $X(k)$ from (4.6) and (4.7) with their estimated, "hatted" versions

$$\hat{A}(k) = \operatorname{diag}\{\hat{v}_0, \ldots, \hat{v}_n, \hat{w}_1, \ldots, \hat{w}_n, \hat{w}_1, \ldots, \hat{w}_n\}, \tag{4.11}$$

and

$$\hat{X}(k) = [\hat{B}_0(k) \cdots \hat{B}_n(k) \quad \hat{B}_0(k-1) \cdots \hat{B}_{n-1}(k-1) \quad \hat{F}_0(k) \cdots \hat{F}_{n-1}(k-1)]^T, \tag{4.12}$$

as is implied by Fig. 3. In terms of adaptive lattice equations analogous to (4.1)–(4.5), we have

$$\hat{y}(k) = \sum_{i=0}^{N} \hat{v}_i(k)\hat{B}_i(k) \tag{4.13}$$

$$\hat{B}_i(k) = \hat{B}_{i-1}(k-1) + \hat{w}_i(k)\hat{F}_{i-1}(k), \quad i = 1, \ldots, N \tag{4.14}$$

$$\hat{F}_i(k) = \hat{F}_{i+1}(k) - \hat{w}_{i+1}(k)\hat{B}_i(k-1), \quad i = 0, \ldots, N-1 \tag{4.15}$$

$$\hat{F}_N(k) = u(k) \tag{4.16}$$

$$\hat{B}_0(k) = \hat{F}_0(k). \tag{4.17}$$

Figure 11 is the block diagram for (4.13)–(4.17) when $N = 3$; it may also be obtained by replacing parameters and signals in Fig. 10 with their "hatted" counterparts.

In order to write down the adaptive algorithm for the lattice [corresponding to (2.15)], we need only interpret the specification of $\hat{\psi}_i(k)$ from Fig. 9 for each i. Notice from Fig. 9 that $\psi_i(k)$ is generated as follows. We have elements $X_j(k)$ of $X(k)$ appearing as inputs to a copy of the adapted system of Fig. 3, with these inputs appearing at points corresponding to where $-\hat{\alpha}_i(k)\hat{X}_j(k)$ enters the original adapted system. Only elements of $\hat{X}(k)$ corresponding to $\hat{\alpha}_i$ appear, and $u \equiv 0$ in the construction of $\hat{\psi}_i(k)$. Then $\hat{\psi}_i(k)$ appears at the output of the system duplicate.

For the $\hat{w}_i(k)$ parameters, which have a rank-2 dependence, the relevent elements of $\hat{X}(k)$ are $\hat{B}_{i-1}(k-1)$ and $\hat{F}_{i-1}(k)$, which would then enter a copy of the lattice at the points where $-\hat{w}_i(k)\hat{B}_{i-1}(k-1)$ and $-\hat{w}_i(k)\hat{F}_{i-1}(k)$ appear in the adapted lattice, respectively. For the third order lattice filter, this is depicted in Fig. 12. Notice that one must keep track of the internal variables $c_{i,j}(k)$ and $d_{i,j}(k)$ in this lattice [which correspond to $B_i(k)$ and $F_i(k)$ in the original lattice of Fig. 10] in order to calculate $\hat{\psi}_{\hat{w}_i}(k)$, which is the lattice output. Equations which propagate the sensitivity functions $\hat{\psi}_{\hat{w}_i}(k)$ for $\hat{w}_i(k)$ are then

$$c_{i,j}(k) = \begin{cases} c_{i,j-1}(k-1) + \hat{w}_j(k)d_{i,j-1}(k), & i \neq j, \quad j = 1, \ldots, N \\ c_{i,j-1}(k-1) - \hat{F}_{j-1}(k) + \hat{w}_j(k)d_{i,j-1}(k), & i = j, \quad j = 1, \ldots, N \\ d_{i,0}(k), & j = 0 \end{cases} \tag{4.18}$$

$$d_{i,j}(k) = \begin{cases} d_{i,j+1}(k) - \hat{w}_{j+1}(k)c_{i,j}(k-1), & i \neq j+1, \quad j = 0, \ldots, N-1 \\ d_{i,j+1}(k) + \hat{B}_j(k-1) - \hat{w}_{j+1}(k)c_{i,j}(k-1), & i = j+1, \quad j = 0, \ldots, N-1 \\ 0, & j = N, \end{cases} \tag{4.19}$$

with

$$\hat{\psi}_{\hat{w}_i}(k) = \sum_{j=0}^{N} \hat{v}_j(k)c_{i,j}(k). \tag{4.20}$$

For the $\hat{v}_i(k)$ parameters, which have a rank-1

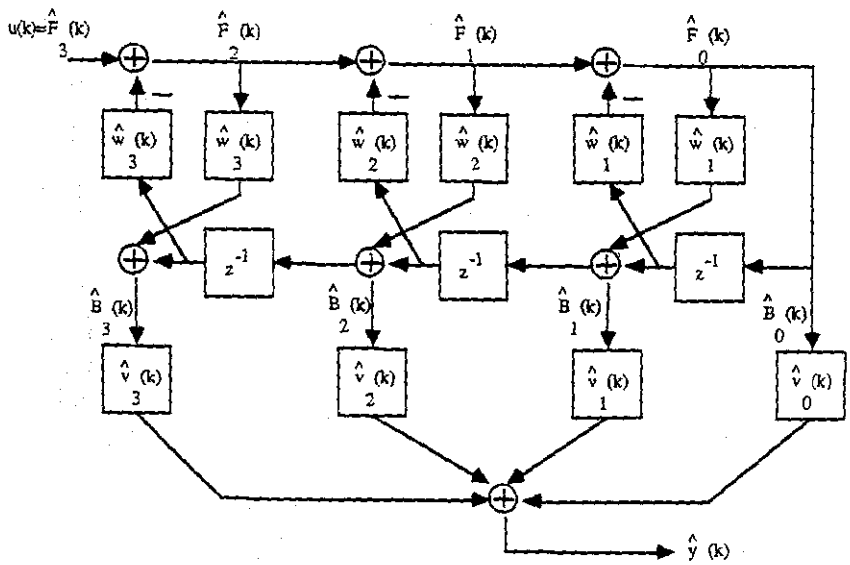

FIG. 11. Adaptive tapped IIR lattice.

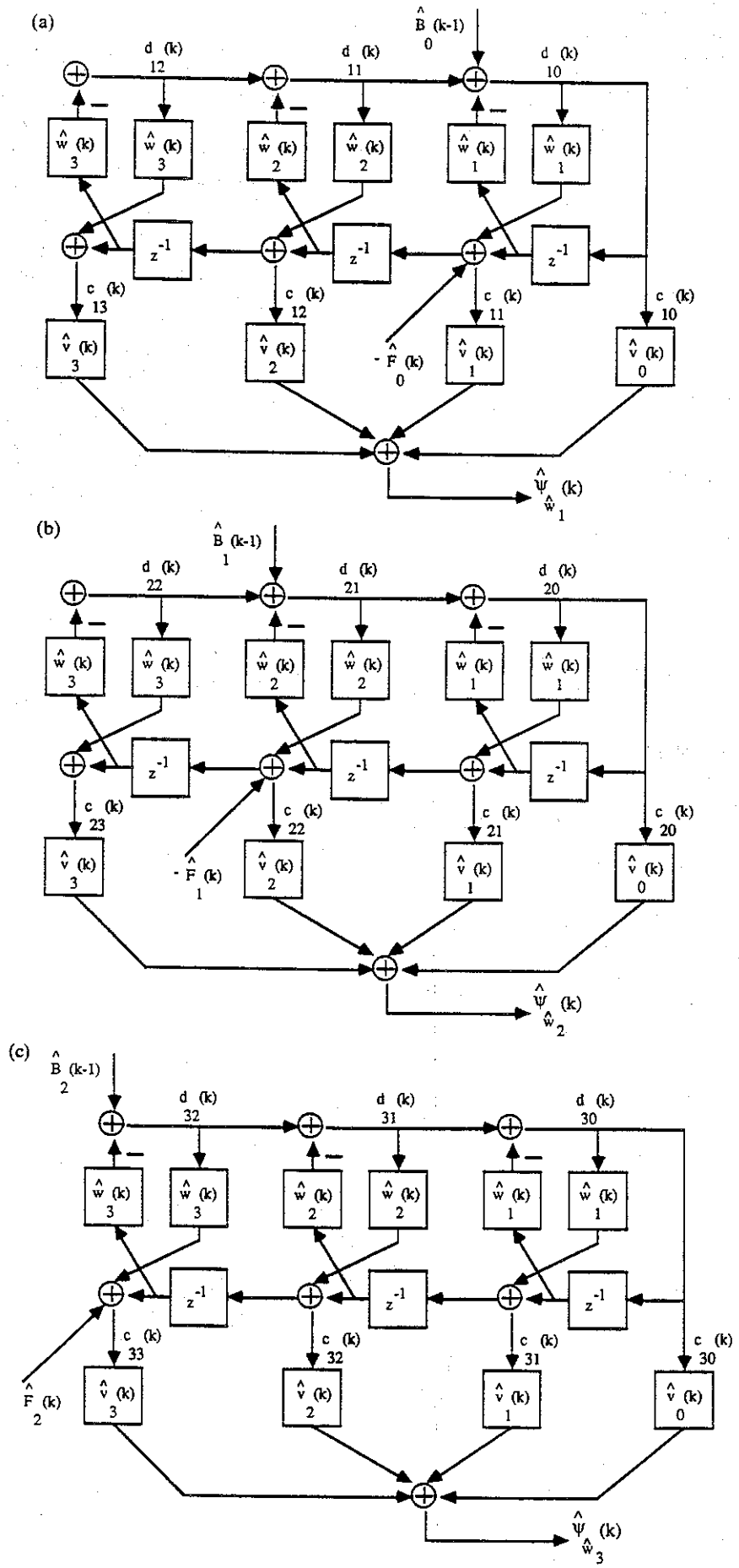

FIG. 12. Approximate sensitivity function generation, (a) $\hat{\psi}_{w_1}(k)$, (b) $\hat{\psi}_{w_2}(k)$ and (c) $\hat{\psi}_{w_3}(k)$.

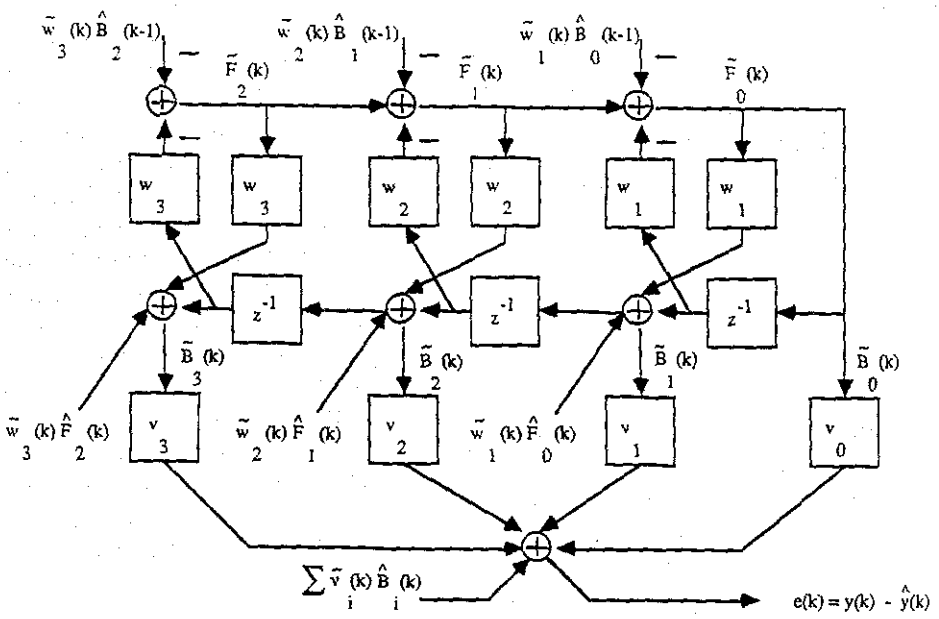

FIG. 13. Prediction error generation.

dependence, the calculation of the sensitivity function $\hat{\psi}_{\hat{v}_i}(k)$ is much simpler. The relevant element of $\hat{X}(k)$ is $\hat{B}_i(k)$, and $-\hat{v}_i(k)\hat{B}_i(k)$ enters the lattice of Fig. 11 only directly at the output (with a factor of -1). Hence,

$$\hat{\psi}_{\hat{v}_i}(k) = -\hat{B}_i(k). \tag{4.21}$$

We may now use (4.21) and (4.18)–(4.20) to specify the algorithm updating the adapted parameters as

$$\hat{v}_i(k+1) = \hat{v}_i(k) - \mu\hat{\psi}_{\hat{v}_i}(k)e(k), \quad i = 0, \ldots, N,$$
$$\hat{w}_i(k+1) = \hat{w}_i(k) - \mu\hat{\psi}_{\hat{w}_i}(k)e(k), \quad i = 1, \ldots, N. \tag{4.22}$$

This algorithm is essentially equivalent to the algorithm proposed by Parikh *et al.* (1980), aside from some minor issues regarding normalization of the adaptation step size. We have commented in Section 2 about interpretation of (4.22) as an "approximate gradient descent"; for comparative purposes, we have reproduced the derivation of the algorithm in Parikh *et al.* (1980) in the Appendix, indicating that by directly calculating the gradient one achieves the same results given here.

We may interpret the prediction error $e(k) = y(k) - \hat{y}(k)$ for the adaptive lattice via Fig. 4(a) as shown in Fig. 13 for the third order case. In Fig. 13, one can justify the labeling of the internal signals as $\tilde{B}_i(k) = B_i(k) - \hat{B}_i(k)$ and $\tilde{F}_i(k) = F_i(k) - \hat{F}_i(k)$ by subtracting (4.14)–(4.17) from (4.2)–(4.5). Note also that Fig. 13 indicates an interpretation of the prediction error $e(k)$ as

$$e(k) = \sum_{i=0}^{N} \tilde{v}_i(k)\hat{B}_i(k) + \sum_{i=1}^{N} G_i(q^{-1})[\tilde{w}_i(k)\hat{F}_{i-1}(k)] + \sum_{i=1}^{N} H_i(q^{-1})[\tilde{w}_i(k)\hat{B}_{i-1}(k-1)], \tag{4.23}$$

where $G_i(q^{-1})$ and $H_i(q^{-1})$ are transfer function operators relating inputs at the various summers in the lattice to the lattice output. Thus, the prediction error is a sum of filtered products of regressor and parameter error terms, where the filtering operations may be different for each product term. This is an example of a composite error as described in Williamson and Johnson (1988).

To describe the error system for the lattice, think of $\hat{v}_i(k)$ and $\hat{w}_i(k)$ as constant in (4.23). Then the parameter error terms may be pulled outside the transfer functions, and (4.23) becomes

$$e(k) = \psi^{\mathrm{T}}(k)\tilde{\theta}(k), \tag{4.24}$$

where $\tilde{\theta}(k)$ is the vector whose entries are $\tilde{v}_i(k)$ and $\tilde{w}_i(k)$. [Compare Fig. 4(a) with Fig. 5.] (4.22) can then be written in terms of $\theta(k)$ as

$$\tilde{\theta}(k+1) \approx [I - \mu\hat{\psi}(k)\psi^{\mathrm{T}}(k)]\tilde{\theta}(k), \tag{4.25}$$

where the approximation is due to the fact that $\tilde{\theta}(k)$ is slowly varying as opposed to constant in the formulation of the prediction error expression (4.24). When $\tilde{\theta}(k) \approx 0$, both $\psi(k)$ and $\hat{\psi}(k)$ in (4.25) will be approximately equal to $\psi^*(k)$. Then

$$\tilde{\theta}(k+1) = [I - \mu\psi^*(k)\psi^{*\mathrm{T}}(k)]\tilde{\theta}(k) \tag{4.26}$$

approximates (4.25).

Most important for the algorithm (4.22) and its error system (4.25) is the local stability result of Section 3. As with the connection of (3.3) and (3.2), the exponential stability of (4.26) for persistently exciting $\psi^*(k)$ establishes the local stability of (4.25) about $\hat{\theta} \approx \theta$. This persistent excitation condition of $\psi^*(k)$ is established if $\psi^*(k)$ is bounded and possesses a sustained

adequate spectral richness. To help maintain the boundedness of $\hat{\psi}(k)$ in (4.25), we keep $|\hat{w}_i(k)| < 1$ for all k by ignoring updates which specify $|\hat{w}_i(k+1)| \geq 1$. This is the crucial stability check required in the algorithm [see the comments following (3.5)]. Here, the lattice form makes this stability check trivial.

The exponential local stability of (4.26) also implies robust local stability for the algorithm (Anderson *et al.*, 1986). We thus expect that local convergence will be maintained, for instance, under output measurement noise. We demonstrate this robustness property in the simulations of Section 5.

For the adaptive lattice, we may comment further on the required persistent excitation condition. This condition can be ensured in the following way. Suppose that $u(k)$ is a linear combination of at least $N+1$ independent sinusoidal signals. Now, if two stable transfer functions both of degree N have the same output for such an input $u(k)$, the two transfer functions are necessarily equal. Therefore, they will have the same coefficients appearing in a lattice realization, due to the uniqueness of the direct to tapped lattice model conversion.

Now lack of persistence of excitation of $\psi^*(k)$ when $u(k)$ is a linear combination of at least $N+1$ distinct sinusoids would imply the existence of a particular nonzero $\tilde{\theta}_p$ for which $\psi^{*T}(k)\tilde{\theta}_p = 0$ for all k. But this would in turn imply that $e(k) \equiv 0$ when the initial parameter error vector $\tilde{\theta}(0)$ is $\tilde{\theta}_p$. By the argument in the preceding paragraph, the transfer functions defined by the initial parameter vector estimate and the true parameter vector would then be equal, so that $\tilde{\theta}_p = 0$, a contradiction. Thus, adequate excitation in $u(k)$ implies persistency of excitation in $\psi^*(k)$.

The above argument establishes local stability for our lattice algorithm [and hence for the algorithm of Parikh *et al.* (1980)] when the input contains at least $N+1$ sinusoids. We have essentially translated a condition granting persistent excitation for the direct form algorithm to a condition granting persistent excitation for the lattice algorithm, by appealing to the one-to-one relationship between the corresponding parametrizations. This tactic may also be used to translate some global stability properties. The one-to-one relationship of direct form and lattice parametrizations enables a preservation of the main qualitative features of the corresponding "error surfaces" (Nayeri and Jenkins, 1989). For example, when the order of the adapted lattice exactly matches that of the lattice being identified and when the input is white, the only local minimum of the associated error surface occurs where $\hat{v}_i = v_i$ and $\hat{w}_i = w_i$ (Söderström and Stoica, 1982). Therefore, if the approximate gradient algorithm accurately descends this error surface, we would expect convergence to either this unique local minimum, or to a point on the stability boundary where the $\hat{w}_i = w_i$ parameters are frozen by the stability check procedure.

Note also that the algorithm here is not the only possible candidate for updating the lattice parameters. Recursive Least Squares (RLS) or Gauss Newton (GN) algorithms have been considered, and they may possess faster convergence rates than the approximate gradient descent (Shynk, 1988). However, as with RLS algorithms for direct form models, the computational complexity of the RLS version of the lattice algorithm is substantially increased when compared to the algorithm presented here.

5. SIMULATION EVIDENCE

This section provides readily reproducible simulated evidence of the local stability and robustness of this locally stable adaptive IIR tapped lattice identifier. Consider the second order lattice of (4.1)–(4.5), with $N = 2$, parametrized by $v_0 = 0.55668$, $v_1 = 0.39210$, $v_2 = 0.89000$, $w_1 = -0.71200$, and $w_2 = 0.25000$, and driven by

$$u(k) = 0.4[1 + \cos(0.3k) + \cos(0.5k) + \cos(0.8k) + \cos(1.0k) + \cos(1.3k)], \quad k > 0, \quad (5.1)$$

starting from zero initial conditions at $k = 0$. Starting the identifier of (4.22) with $\hat{v}_i(0) = \hat{w}_i(0) = \hat{B}_i(0) = \hat{F}_i(0) = 0$ for all i, and setting $\mu = 0.01$, yields the decaying prediction error $y - \hat{y}$ in Fig. 14(a) and the linearly decreasing logarithm of the summed squared parameter error $\log_{10}[\tilde{v}_0^2(k) + \tilde{v}_1^2(k) + \tilde{v}_2^2(k) + \tilde{w}_1^2(k) + \tilde{w}_2^2(k)]$ in Fig. 14(b). Note that, as expected, the prediction error and summed squared parameter error are converging to zero, with the summed squared parameter error doing so exponentially fast.

The expected robustness to system output measurement noise can be tested by adding a low-variance, white, zero-mean noise to the prediction error $y - \hat{y}$ before it is used in the algorithm in (4.22). Figure 15(a) plots the prediction error before this noise is added, which is the disturbance-free prediction error we are unable to measure. Figure 15(b) plots the summed squared parameter error which shows an initial exponential decay that quickly reaches a floor that represents the unremovable parameter variation due to the unpredictable noise in the prediction error and a nonvanishing adaptive algorithm step size. Our understanding of (3.1)

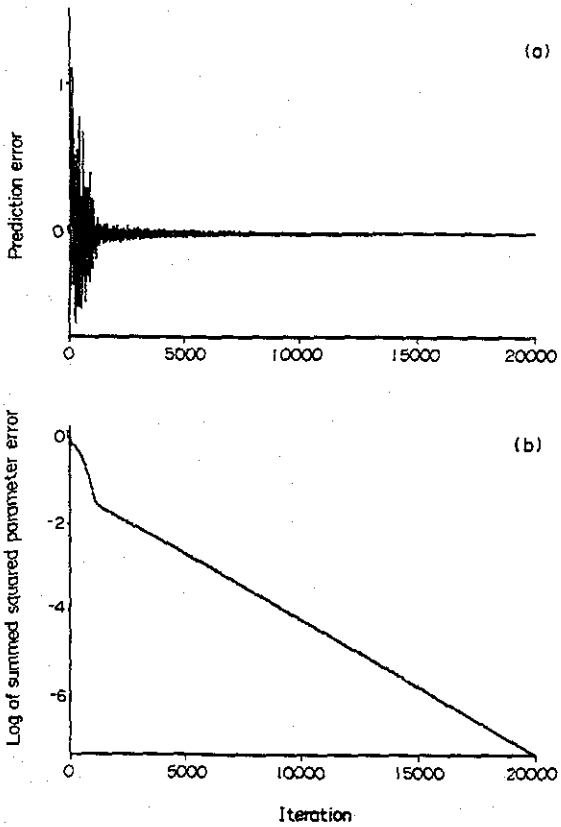

FIG. 14. Local asymptotically convergent behavior in ideal case, $\mu = 0.01$.

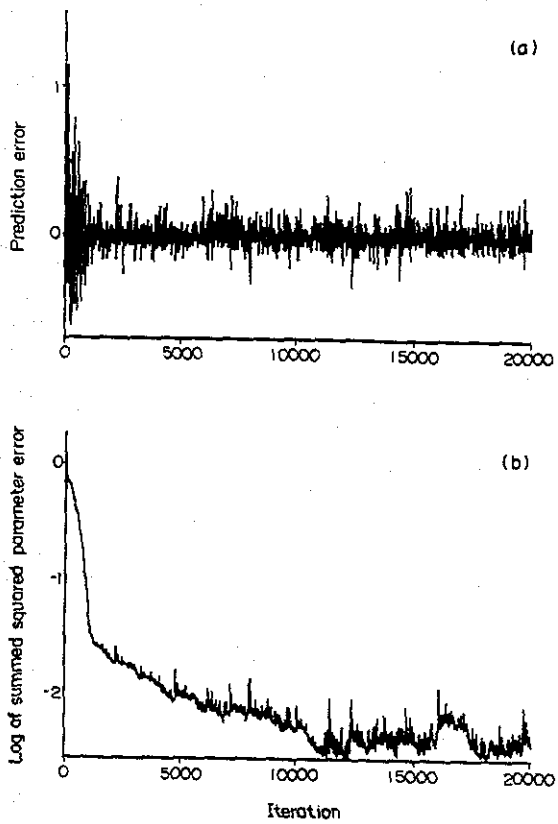

FIG. 15. Robust behavior despite output measurement noise, $\mu = 0.01$.

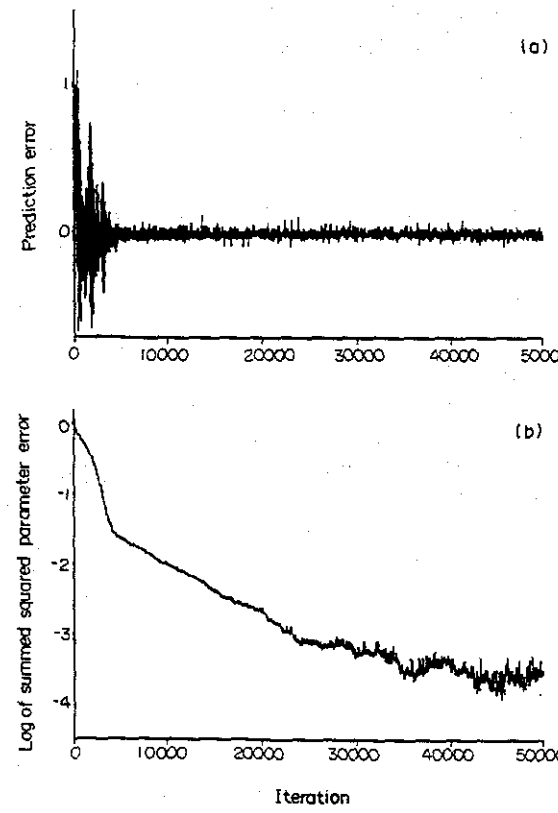

FIG. 16. Robust behavior despite output measurement noise, $\mu = 0.003$.

suggests that decreasing the step size in (4.22), say from 0.01 to 0.003, should result in a slower convergence rate during the exponential decay phase and a lower floor to the unremovable variance in the summed squared parameter error. This is verified by comparing Figs 15 and 16.

One should note that our simulations employed a simple stability check and projection procedure: if the calculated new value of any $\hat{w}_i(k+1)$ in (4.22) had magnitude greater than or equal to one, then that update was ignored, leaving $\hat{w}_i(k+1) = \hat{w}_k(k)$ for that particular value of i. For the simulations conducted starting from zero initial conditions, the need for the stability check did not arise. However, in other simulations we conducted, in which the $|\hat{w}_i(0)|$ were initialized nearer to unit values, the stability check and projection procedure prevented unstable behavior which occurred when the same simulation was conducted without the stability check.

6. CONCLUSIONS

We have in this paper developed a recursive identification scheme for linear systems representable by a Feedback Gain Model. The error system formulation for the adaptive algorithm is in a form amenable to analytical tools which

have been extracted from nonlinear system stability theory to prove the local exponential stability and robustness of a variety of adaptive parameter estimators (Anderson *et al.*, 1986; Sethares *et al.*, 1989; Anderson and Johnson, 1988; Johnson, 1988). Thus, the output error identifier is provably locally robust under certain conditions, such as persistent excitation similar to that associated with other output error identifiers (Johnson, 1984; Anderson *et al.*, 1986; Johnson, 1988). Simulations in Section 5 verify the existence of such behavior for an adaptive lattice identifier.

We have also applied the Feedback Gain Model identifier in the development of an adaptive algorithm for the identification of the parameters in a tapped lattice model for a linear system. This development is a reinterpretation of an adaptive IIR filter algorithm developed by Parikh *et al.* (1980) as an algorithm for output error identification that enjoys a computationally trivial stability check and maintenance procedure. The stability analysis for the general FGM identifier applies to the lattice model, and it provides the algorithm of Parikh *et al.* (1980) with its first formal convergence proof. The general framework of the Feedback Gain Model also has application in other identification problems. For instance, the identification of linear systems using parallel and cascade models (Nayeri and Jenkins, 1989; Shynk, 1989; Williamson and Johnson, 1989) fits in this model class, so that the stability and robustness results developed here apply in these cases as well.

Several issues remain to be addressed if this algorithm is to be promoted as a viable adaptive output error identifier. Theoretically and practically, description of the size of attraction of the local robustness region in the space of initial parameter estimates and other initial conditions about a satisfactory answer is a significant concern. Establishing an accurate approximation for the local convergence rate is also a nontrivial task. These, and other, questions are rather standard accoutrements of the stability theory approach to adaptive system analysis that provide insight into the applicability and performance of adaptive algorithms in various problems. Such studies of applicability and performance of this adaptive algorithm are now in order due to our establishment of its local stability.

One particular issue of concern in practical applications is the computational burden of the algorithm implementation. The calculation and storage of the N (approximate) sensitivity functions $\hat{\psi}_i$ results in a potentially large computational load for the algorithm of (2.15). However, results from Bingulac *et al.* (1988) may be useful in reducing this burden by using one sensitivity model from which all the needed sensitivity functions may be determined.

This adaptive tapped lattice IIR filter also serves as an example of the extension of adaptive parameter estimator error system analysis to situations where the prediction error can be approximated as the inner product of the parameter error vector and a regressor that has been passed through a matrix transfer function. Recall (4.23), which reveals the prediction error for the lattice identifier to be a composite of differently filtered products of parameter error and regressor terms. This structure contrasts with the more familiar prediction error form consisting of an inner product of the regressor and parameter error vectors passed through a scalar transfer function [as in Sethares *et al.* (1989)]. The added complexity of the compositive error strongly influences issues pertaining to algorithm selection (Williamson *et al.*, 1991). Other examples of such composite error systems exist in quite different adaptive system applications (Williamson and Johnson, 1988), and we expect that a number of systems in the Feedback Gain Model class will possess composite errors as well.

REFERENCES

Anderson, B. D. O., R. R. Bitmead, C. R. Johnson, Jr., P. V. Kokotovic, R. L. Kosut, I. M. Y. Mareels, L. Praly and B. D. Riedle (1986). *Stability of Adaptive Systems: Passivity and Averaging Analysis*. MIT Press, Cambridge, MA.

Anderson, B. D. O. and C. R. Johnson, Jr (1988). Identification of linear systems containing unknown linear gain elements. Proc. 8th IFAC Symp. on Identification and System Parameter Estimation, Beijing, China.

Bingulac, S., J. H. Chow and J. R. Winkelman (1988). Simultaneous generation of sensitivity functions—transfer function matrix approach. *Automatica*, **24**, 239–242.

Dasgupta, S., B. D. O. Anderson and R. J. Kaye (1984). Identification of economically parametrized systems. *Proc. 9th IFAC World Congress*. Budapest, Hungary, X, 96–100.

Dasgupta, S., B. D. O. Anderson and R. J. Kaye (1986). Output error identification methods for partially known systems. *Int. J. Control*, **43**, 177–191.

Desoer, C. A. and M. Vidyasagar (1975). *Feedback Systems: Input-Output Properties*. Academic Press, New York.

Friedlander, B. (1982). Lattice filters for adaptive processing. *Proc. IEEE*, **70**, 829–867.

Gray, A. H., Jr and J. D. Markel (1973). Digital lattice and ladder filter synthesis. *IEEE Trans. Audio Electroacoust.*, **AU-21**, 491–500.

Hahn, W. (1967). *Stability of Motion*. Springer, New York.

Johnson, C. R., Jr (1984). Adaptive IIR filtering: Current results and open issues. *IEEE Trans. Inform. Theory* **IT-30**, 237–250.

Johnson, C. R., Jr (1988). *Lectures on Adaptive Parameter Estimation*. Prentice-Hall, Englewood Cliffs, NJ.

Kokotovic, P. V., J. V. Medanic, M. I. Vuskovic and S. P. Bingulac (1966). Sensitivity method in the experimental design of adaptive control systems. *Proc. 3rd IFAC Congress*. London, **I**, 45B.1–12.

Landau, I. D. (1976). Unbiased recursive identification using

model reference adaptive techniques. *IEEE Trans. Aut. Control.* **AC-21**, 194–202.
Ljung, L. and T. Söderström (1983). *Theory and Practice of Recursive Identification.* MIT Press, Cambridge, MA.
Medanic, J. V. and P. Kokotovic (1965). Some problems in the development of adaptive systems using the sensitivity operator. *Proc. 2nd IFAC Symp. on the Theory of Self-Adaptive Control Systems.* Teddington, UK, 204–212.
Narendra, K. S., A. M. Annaswamy and R. P. Singh (1985). A general approach to the stability analysis of adaptive systems. *Int. J. Control*, **41**, 193–216.
Nayeri, M. and W. K. Jenkins (1989). Alternate realizations of adaptive IIR filters and properties of their performance surfaces. *IEEE Trans. Circuits Syst.*, **CAS-36**, 485–496.
Parikh, D., N. Ahmed and S. Stearns (1980). An adaptive lattice algorithm for recursive filters. *IEEE Trans. Acoust. Speech Signal Process.*, **ASSP-28**, 110–111.
Sethares, W. A., B. D. O. Anderson and C. R. Johnson, Jr (1989). Adaptive algorithms with filtered regressor and filtered error. *Math. Control Signals Syst.*, **2**, 381–403.
Shynk, J. J. (1988). On lattice-form algorithms for adaptive IIR filtering. *Proc. ICASSP.* New York, 1554–1557.
Shynk, J. J. (1989). Adaptive IIR filtering using parallel-form realizations. *IEEE Trans. Acoust. Speech Signal Process.*, **ASSP-37**, 519–533.
Söderström, T and P. Stoica (1982). Some properties of the output error method. *Automatica*, **18**, 93–99.
Vidyasagar, M. (1978). *Nonlinear Systems Analysis.* Prentice-Hall, Englewood Cliffs, NJ.
Weiss, A. and D. Mitra (1979). Digital adaptive filters: conditions for convergence, rates of convergence, effects of noise and errors arising from implementation. *IEEE Trans. Inform. Theory* **IT-25**, 637–652.
Widrow, B. and S. Stearns (1985). *Adaptive Signal Processing.* Prentice-Hall, Englewood Cliffs, NJ, 164–182.
Williamson, G. A. and C. R. Johnson, Jr (1988). On a generalized structure for new error systems in discrete adaptive parameter estimators. *27th IEEE Conf. on Decision and Control.* Austin, TX, 769–774.
Williamson, G. A. and C. R. Johnson, Jr (1989). Excitation-dependent stability for nongradient parallel-form adaptive IIR filters. *Proc. 23rd Asilomar Conf. on Signals, Systems and Computers.* Pacific Grove, CA, 552–556.
Williamson, G. A., B. D. O. Anderson and C. R. Johnson, Jr (1991). On the local stability properties of adaptive parameter estimators with composite errors and split algorithms. *IEEE Trans. Aut. Control* **AC-36**, 463–473.

APPENDIX: DERIVATION OF LATTICE GRADIENT DESCENT ALGORITHM

In this Appendix we derive an approximate gradient descent algorithm for updating the parameters in the adapted lattice model of (4.13)–(4.17). The derivation follows that of Parikh *et al.* (1980), and it leads to an algorithm identical to that given by (4.18)–(4.22).

The approximate gradient descent procedure for the lattice model is given by

$$\hat{v}_i(k+1) = \hat{v}_i(k) - \mu \frac{\partial \frac{1}{2}\{[y(k) - \hat{y}(k)]^2\}}{\partial \hat{v}_i(k)} \tag{A.1}$$

$$\hat{w}_i(k+1) = \hat{w}_i(k) - \mu \frac{\partial \frac{1}{2}\{[y(k) - \hat{y}(k)]^2\}}{\partial \hat{w}_i(k)}. \tag{A.2}$$

Since $y(k)$ is not a function of $\hat{v}_i(k)$ or $\hat{w}_i(k)$, (A.1) and (A.2) become

$$\hat{v}_i(k+1) = \hat{v}_i(k) + \mu[y(k) - \hat{y}(k)] \frac{\partial \hat{y}(k)}{\partial \hat{v}_i(k)} \tag{A.3}$$

$$\hat{w}_i(k+1) = \hat{w}_i(k) + \mu[y(k) - \hat{y}(k)] \frac{\partial \hat{y}(k)}{\partial \hat{w}_i(k)}. \tag{A.4}$$

From (4.13)

$$\frac{\partial \hat{y}(k)}{\partial \hat{v}_i(k)} = \hat{B}_i(k) + \sum_{j=0}^{N} \hat{v}_j(k) \frac{\partial \hat{B}_j(k)}{\partial \hat{v}_i(k)} = \hat{B}_i(k), \tag{A.5}$$

since the $\hat{B}_j$ do not depend explicitly on the $\hat{v}_i$. Also, from (4.13)

$$\frac{\partial \hat{y}(k)}{\partial \hat{w}_i(k)} = \sum_{j=0}^{N} \hat{v}_j(k) \frac{\partial \hat{B}_j(k)}{\partial \hat{w}_i(k)}. \tag{A.6}$$

From (4.14)

$$\frac{\partial \hat{B}_j(k)}{\partial \hat{w}_i(k)} = \frac{\partial \hat{B}_{j-1}(k-1)}{\partial \hat{w}_i(k)} + \frac{\partial \hat{w}_j(k)}{\partial \hat{w}_i(k)} \hat{F}_{j-1}(k) + \hat{w}_j(k) \frac{\partial \hat{F}_{j-1}(k)}{\partial \hat{w}_i(k)}. \tag{A.7}$$

Now we make an approximation, predicated on a slow rate of parameter adaptation, which enables a recursive calculation of the partial derivative of (A.7). Under the assumption that μ in (A.1) and (A.2) is sufficiently small so that

$$\frac{\partial \hat{B}_{j-1}(k-1)}{\partial \hat{w}_i(k)} \approx \frac{\partial \hat{B}_{j-1}(k-1)}{\partial \hat{w}_i(k-1)} \tag{A.8}$$

makes sense, we can approximate (A.7) by

$$\frac{\partial \hat{B}_j(k)}{\partial \hat{w}_i(k)} \approx \begin{cases} \dfrac{\partial \hat{B}_{j-1}(k-1)}{\partial \hat{w}_i(k-1)} + \hat{w}_j(k) \dfrac{\partial \hat{F}_{j-1}(k)}{\partial \hat{w}_i(k)}, & i \neq j \\ \dfrac{\partial \hat{B}_{j-1}(k-1)}{\partial \hat{w}_i(k-1)} + \hat{F}_{j-1}(k) + \hat{w}_j(k) \dfrac{\partial \hat{F}_{j-1}(k)}{\partial \hat{w}_i(k)}, & i = j \end{cases} \tag{A.9}$$

for $j = 1, \ldots, N$. Note the (time) recursive form of (A.9). From (4.15)

$$\begin{aligned} \frac{\partial \hat{F}_j(k)}{\partial \hat{w}_i(k)} &= \frac{\partial \hat{F}_{j+1}(k)}{\partial \hat{w}_i(k)} - \frac{\partial \hat{w}_{j+1}}{\partial \hat{w}_i(k)} \hat{B}_j(k-1) \\ &\quad - \hat{w}_{j+1}(k) \frac{\partial \hat{B}_j(k-1)}{\partial \hat{w}_i(k)} \\ &\approx \begin{cases} \dfrac{\partial \hat{F}_{j+1}(k)}{\partial \hat{w}_i(k)} - \hat{w}_{j+1}(k) \dfrac{\partial \hat{B}_j(k-1)}{\partial \hat{w}_i(k-1)}, & i \neq j+1 \\ \dfrac{\partial \hat{F}_{j+1}(k)}{\partial \hat{w}_i(k)} - \hat{B}_j(k-1) - \hat{w}_{j+1}(k) \\ \quad \times \dfrac{\partial \hat{B}_j(k-1)}{\partial \hat{w}_i(k-1)}, & i = j+1 \end{cases} \end{aligned} \tag{A.10}$$

for $j = 0, \ldots, N-1$. From (4.16),

$$\frac{\partial \hat{F}_N(k)}{\partial \hat{w}_i(k)} = 0. \tag{A.11}$$

From (4.17),

$$\frac{\partial \hat{B}_0(k)}{\partial \hat{w}_i(k)} = \frac{\partial \hat{F}_0(k)}{\partial \hat{w}_i(k)}. \tag{A.12}$$

Consider the two two-dimensional arrays, indexed by $i = 1, \ldots, N$ and $j = 0, \ldots, N$, defined by the following recursive equations:

$$\bar{c}_{i,j}(k) = \begin{cases} \bar{c}_{i,j-1}(k-1) + \hat{w}_j(k)\bar{d}_{i,j-1}(k), \\ \quad i \neq j, \quad j = 1, \ldots, N \\ \bar{c}_{i,j-1}(k-1) + \hat{F}_{j-1}(k) + \hat{w}_j(k)\bar{d}_{i,j-1}(k), \\ \quad i = j, \quad j = 1, \ldots, N \\ \bar{d}_{i,0}(k), \quad j = 0 \end{cases} \tag{A.13}$$

$$\bar{d}_{i,j}(k) = \begin{cases} \bar{d}_{i,j+1}(k) - \hat{w}_{j+1}(k)\bar{c}_{i,j}(k-1), \\ \quad i \neq j+1, \quad j = 0, \ldots, N-1 \\ \bar{d}_{i,j+1}(k) - \hat{B}_j(k-1) - \hat{w}_{j+1}(k)\bar{c}_{i,j}(k-1), \\ \quad i = j+1, \quad j = 0, \ldots, N-1 \\ 0, \quad j = N \end{cases} \tag{A.14}$$

where initial conditions are arbitrary. Compare (A.13) and (A.14) to (A.9)–(A.12). Notice that $\bar{c}_{i,j}(k)$ and $\bar{d}_{i,j}(k)$ satisfy exact versions of the equations approximately satisfied by $\partial \hat{B}_j(k)/\partial \hat{w}_i(k)$ and $\partial \hat{F}_j(k)/\partial \hat{w}_i(k)$. Furthermore, the equations generating $\bar{c}_{i,j}(k)$ and $\bar{d}_{i,j}(k)$ define a lattice whose

homogeneous part is the same as that of the adapted lattice of Fig. 11, which we presume is stable. Hence any initial condition effects will be forgotten, and we can therefore regard $\bar{c}_{i,j}(k)$ and $\bar{d}_{i,j}(k)$ as approximations of $\partial\hat{B}_j(k)/\partial\hat{w}_i(k)$ and $\partial\hat{F}_j(k)/\partial\hat{w}_i(k)$, respectively. With (A.8) in mind, we note that the quality of the approximation will improve as the size of the adaptive gain μ decreases.

The adaptive algorithm of Parikh *et al.* (1980) uses (A.5) and the fact that $\bar{c}_{i,j}(k)$ approximates $\partial\hat{B}_j(k)/\partial\hat{w}_i(k)$ to replace (A.3) and (A.4) with

$$\hat{v}_i(k+1) = \hat{v}_i(k) + \mu_i[y(k) - \hat{y}(k)]\hat{B}_i(k), \quad i = 0, \ldots, N \tag{A.15}$$

$$\hat{w}_i(k+1) = \hat{w}_i(k) + \rho_i(k)[y(k) - \hat{y}(k)]\left[\sum_{j=0}^{N} \hat{v}_j(k)\bar{c}_{i,j}(k)\right], \quad i = 1, \ldots, N. \tag{A.16}$$

(A.15) and (A.16), together with (A.13) and (A.14), therefore define an approximate gradient descent algorithm for updating the parameters of the adapted lattice.

Having derived the algorithm of Parikh *et al.* (1980), let us show that it is the same as the algorithm developed in Section 4 [equation (4.22) with (4.18)–(4.21)]. First notice from (4.21) and the interpretation of $\hat{\psi}_{\hat{v}_i}$ as the negative of the partial of $\hat{y}(k)$ with respect to $\hat{v}_i$ that the updates of the $\hat{v}_i$ parameters are identical. To see that the $\hat{w}_i$ updates are the same, compare (A.13) and (A.14) to (4.18) and (4.19). If we identify $\bar{c}_{i,j}$ with $c_{i,j}$ and $\bar{d}_{i,j}$ with $d_{i,j}$, then these pairs of equations are identical, except for the fact that $\hat{F}_{j-1}(k)$ appears with a minus sign in (4.18) and with a plus sign in (A.13), and that $\hat{B}_j(k-1)$ appears with a plus sign in (4.19) and with a minus sign in (A.14). As these equations may be viewed as describing a lattice whose only external inputs are given by the $\hat{F}_{j-1}(k)$ and $\hat{B}_j(k-1)$ terms (as in Fig. 12), these differences of sign will simply reflect a difference of sign at the output of the lattice. Therefore

$$\sum_{j=0}^{N} \hat{v}_j(k)\bar{c}_{i,j}(k) = -\sum_{j=0}^{N} \hat{v}_j(k)c_{i,j}(k). \tag{A.17}$$

Noting the role of (4.20) in (4.22), and the role of the left hand side of (A.17) in (A.16), we see that the $\hat{w}_i$ updates are identical as well. Therefore, the algorithm embodied by (4.18)–(4.22) is a reinterpretation of the algorithm of Parikh *et al.* (1980).