IFAC Symp. New Delhi, 1982
on Digital Control

C16

D-STEP AHEAD PREDICTION IN LATTICE AND LADDER FORM

V. WERTZ and M. GEVERS

Department of Electrical Engineering, Louvain University

Bâtiment Maxwell, B-1348 Louvain-la-Neuve, Belgium

Abstract. We use the orthogonalizing property of the linear prediction lattice filter to construct a d-step forward and backward prediction filter in lattice and ladder form.
An exact solution is presented first assuming a stationary observation process using orthogonal projections in Hilbert space.
An adaptive implementation using least squares recursions follows, which is a generalization of previous least squares lattice filters for one-step prediction.

I INTRODUCTION

Since the publication of Itakura and Saito's two-multiplier lattice filter [1], lattice filters have been extensively studied and have given rise to a number of applications in linear prediction, communication, signal processing and identification. The main feature of the lattice filter is that it is an orthogonalizing device which replaces the original signal process by a sequence of orthogonal residuals generating the same space. As a result the adaptation of the different stages of a lattice filter can be checked by inspection and, in an adaptive implementation, the reflection coefficients can always be computed in such a way that stability of the overall filter is guaranteed.

Because the lattice filter produces as its output one-step ahead and one-step backward residuals of the incoming observations, it is straightforward to use it as a one-step prediction filter and this has been one of its most obvious applications. As such it is nothing else but a clever implementation of the Levinson algorithm [2]-[3]. We show in this paper that the basic orthogonalizing property of the lattice filter also allows one to produce d-step forward and backward predictions of the observations with little extra computation. The basic idea is to use the sequence of one-step backward residuals as an orthogonal basis for the space of past observations, and to construct the d-step predicted estimates of the signal process $y(t)$ by projecting the future values of $y(t)$ on this basis. Similarly backward d-step predicted estimates of $y(t)$ are constructed by projecting past values of $y(t)$ on the appropriate basis formed by orthogonalized 1-step forward residuals. As can be guessed, our presentation will rely heavily on projection arguments in Hilbert space. We also give "exact" least squares recursions for the adaptive implementation of this d-step predictor. Our derivation is an extension of the L.S. solution due to Morf et al. (See [4]-[5] for the one-step predictor.

In Section 2 we recall the basic two-multiplier lattice strcture of Itakura and Saito together with some of its main properties. We use the orthogonal sequences produced by this basic lattice filter to derive a new d-step lattice filter in Section 3, and we show its similarity with the one-step predictor.

In Section 4, we derive the least squares recursions (order updates and time updates) for the computation of the d-step residuals while we show in Section 5 how to obtain recursive least squares d-step ahead predictions.

II THE BASIC LATTICE STRUCTURE

In this section we first give the equations of the two-multiplier lattice that was introduced by Itakura and Saito [1], and has been further studied by a number of authors [4]-[8]. This lattice structure constitutes a whitening filter and provides an all-pole model of increasing order for the signal process. In addition one-step ahead predictions of increasing order for the signal process are obtained from this lattice form with almost no extra computations. A major advantage of the lattice formulation for the prediction or the modelling of signal processes is that the stability of the lattice filter can be checked by inspection.

We shall consider a real valued (wide-sense) stationary vector random process $y(t)$,

$t \in Z$, $y(t) \in \mathbb{R}^p$, and the Hilbert space H spanned by the components of the $y(t)$'s. We shall denote by Y_t^{t+k} for $k \geq 0$ the closed linear subspace of H spanned by the components of $\{y(t), y(t+1), \ldots, y(t+k)\}$; Y_t^{t+k} for $k < 0$ will denote the empty space. Finally for every element $y(\tau)$ in H, $E\{y(\tau)| Y_t^{t+k}\}$ will denote the orthogonal projection of $y(\tau)$ onto Y_t^{t+k}, i.e.

$$E\{(y(\tau)- E\{y(\tau)| Y_t^{t+k}\})y^T(j)\} = 0,$$
$$j=t,\ldots, t+k.$$

With this definition $E\{y(t)| y(s)\} = E\{y(t)y^T(s)\}E\{y(s)y^T(s)\}^{-1}y(s)$.[1]
In the Gaussian case, $E\{y(t)| y(s)\}$ is the conditional expectation of $y(t)$ given $y(s)$.

We now introduce the following random processes associated with the process $y(t)$.

$$f_k(t) \triangleq y(t) - E\{y(t)| Y_{t-k}^{t-1}\} \qquad (2.1a)$$

$$g_k(t) \triangleq y(t-k) - E\{y(t-k)| Y_{t-k+1}^{t}\} \qquad (2.1b)$$

$$f_0(t) = g_0(t) = y(t) \qquad (2.1c)$$

The variables $f_k(t)$ and $g_k(t)$ are called the forward and backward residuals of order k of the process $y(t)$. It can be shown that they can be computed by the following recursive formulae :

$$f_{k+1}(t) = f_k(t) - K_{k+1}^b g_k(t-1) \qquad (2.2a)$$

$$g_{k+1}(t) = g_k(t-1) - K_{k+1}^f f_k(t) \qquad (2.2b)$$

$$f_0(t) = g_0(t) = y(t) \qquad (2.2c)$$

where

$$K_{k+1}^b = S_k(R_k^b)^{-1}, \; K_{k+1}^f = S_k^T(R_k^f)^{-1} \qquad (2.3)$$

$$S_k = E\{f_k(t)g_k^T(t-1)\} \qquad (2.4)$$

$$R_k^b = E\{g_k(t)g_k^T(t)\}, \; R_k^f = E\{f_k(t)f_k^T(t)\} \qquad (2.5)$$

Equations (2.2) can be implemented as a lattice filter as shown in Figure 1.

[1] We assume that y is a full rank process, so that the inverse exists.

Properties of the Basic Lattice Filter

We list here some properties of the lattice filter which will be used later on.

1. The reflection coefficient matrices K_{k+1}^b and K_{k+1}^f are such that they minimize $\mathrm{tr}R_{k+1}^f$ and $\mathrm{tr}R_{k+1}^b$ respectively.

2. The lattice filter generates an all-zero whitening filter for the $y(t)$-process via the following recursive relations :

$$A_{k+1}(z) = A_k(z) - K_{k+1}^b \, z^{-1} \, B_k(z) \qquad (2.6a)$$

$$B_{k+1}(z) = z^{-1} \, B_k(z) - K_{k+1}^f \, A_k(z) \qquad (2.6b)$$

$$A_0(z) = B_0(z) = I \qquad (2.6c)$$

$A_k(z)$ and $D_k(z)$ are k-th order matrix polynomials :

$$A_k(z) = \sum_{i=0}^{k} A_{k,i} \, z^{-i}, \; B_k(z) = \sum_{i=0}^{k} B_{k,i} \, z^{-i}$$

With $A(z)$ and $B(z)$ defined as above, we have

$$A_k(z)Y(z) = F_k(z), \; B_k(z)Y(z) = G_k(z)$$

where $F_k(z)$ and $G_k(z)$ are the z-transforms of the forward and backward residuals. Note that $A_{k,0} = B_{k,k} = I$, while $A_{k,k} = -K_k^b$ and $B_{k,0} = -K_k^f$.

3. The k-th order forward and backward one-step ahead predictors for y are :

$$E\{y(t+1)| Y_{t-k+1}^{t}\} = - \sum_{i=1}^{k} A_{k,i} y(t-i+1)$$

$$E\{y(t-k)| Y_{t-k+1}^{t}\} = -\sum_{i=0}^{k-1} B_{k,i} \, y(t-i)$$

4. The forward and backward residuals satisfy the following orthogonality properties :

$$E\{f_k(t+k)f_j^T(t+j)\} = R_k^f \, \delta_{kj} \qquad (2.7)$$

$$E\{g_k(t)g_j^T(t)\} = R_k^b \, \delta_{kj} \qquad (2.8)$$

$$E\{g_k(t)y^T(t-i)\} = 0 \qquad 0 \leq i \leq k-1 \qquad (2.9)$$

$$E\{f_k(t)y^T(t-i)\} = 0 \qquad 1 \leq i \leq k \qquad (2.10)$$

Two alternative expressions for S_k (and hence for K_{k+1}) can be derived from (2.9) and (2.10) :

$$S_k = E\{y(t)g_k^T(t-1)\} = E\{f_k(t)y^T(t-k-1)\} \qquad (2.11)$$

It also follows easily from (2.7) - (2.8) and the definitions (2.1) that both

$\{g_0(t), g_1(t), \ldots, g_{k-1}(t)\}$ and

$\{f_0(t-k+1), f_1(t-k+2), \ldots, f_{k-1}(t)\}$ are

orthogonal bases for Y^t_{t-k+1}.

5. Using $\{g_0(t), \ldots, g_{k-1}(t)\}$ as an orthogonal basis for Y^t_{t-k+1} leads to an alternative expression for $E\{y(t+1)| Y^t_{t-k+1}\}$:

$$E\{y(t+1)| Y^t_{t-k+1}\} = E\{y(t+1)| g_0(t), \ldots, g_{k-1}(t)\}$$

$$= \sum_{i=0}^{k-1} E\{y(t+1)| g_i(t)\}$$

$$= \sum_{i=0}^{k-1} E\{f_i(t+1)| g_i(t)\} \quad (2.12)$$

$$= \sum_{i=0}^{k-1} K^b_{i+1} g_i(t) \quad (2.13)$$

All the terms in (2.13) are readily available from the lattice filter of Fig. 1. Therefore the lattice filter is not only a whitening filter and a modelling filter for the $y(t)$ process : with an additional summation (which now constitutes a ladder filter) it also provides a one-step ahead predictor.
A corresponding expression can be obtained for the one-step backward predictor, by projecting $y(t-k)$ on the orthogonal basis $\{f_0(t-k+1),$ $f_1(t-k+2), \ldots, f_{k-1}(t)\}$:

$$E\{y(t-k)| Y^t_{t-k+1}\} = \sum_{i=1}^{k} K^f_i f_{i-1}(t-k+i)$$

III THE D-STEP AHEAD PREDICTOR IN LATTICE AND LADDER FORM

The same idea that was used in the last Section to construct a one-step ahead predictor for $y(t+1)$ from an orthogonal transformation of Y^t_{t-k+1} using the backward residual sequence will be used now to construct a d-step ahead predictor. By the same argument as before we have :

$$E\{y(t+d)| Y^t_{t-k+1}\} = E\{y(t+d)| g_0(t), g_1(t), \ldots, g_{k-1}(t)\}$$

$$= \sum_{i=0}^{k-1} E\{y(t+d)| g_i(t)\} \quad (3.1)$$

$$= \sum_{i=0}^{k-1} K^b_{d,i+1} g_i(t) \quad (3.2)$$

where

$$K^b_{d,i+1} = E\{y(t+d)g_i^T(t)\}(R^b_i)^{-1} \triangleq S_{d,i}(R^b_i)^{-1} \quad (3.3)$$

We now define the <u>d-step forward residual of order k</u> :

$$f_{d,k}(t+d) = y(t+d) - E\{y(t+d)| Y^t_{t-k+1}\} \quad (3.4)$$

With these d-step ahead residuals we can now derive an alternative expression for $S_{d,i}$ which will be similar to (2.4) in the case of one-step ahead predictions. Using the orthogonality conditions (2.9) we have

$$S_{d,i} = E\{y(t+d)g_i^T(t)\} = E\{f_{d,i}(t+d)\ g_i^T(t)\}$$

$$= E\{f_{d,i}(t+d)y^T(t-i)\} \quad (3.5)$$

Finally, we derive a recursive relation for $f_{d,k}(t)$, where the recursion is on k, the order of the predictor. From (3.4) and (3.2) we have :

$$f_{d,k+1}(t) = y(t) - E\{y(t)| Y^{t-d}_{t-d+k}\}$$

$$= y(t) - \sum_{i=0}^{k} K^b_{d,i+1} g_i(t-d) \quad (3.6)$$

Hence

$$f_{d,k+1}(t) = f_{d,k}(t) - K^b_{d,k+1} g_k(t-d) \quad (3.7)$$

with initial condition

$$f_{d,0}(t) = y(t) \quad (3.9)$$

Note the similarity between (3.7) and (2.2a) ; for d=1, these equations become identical. The d-step ahead predictor in lattice and ladder form is presented in Fig. 2. The basic lattice filter is used to generate the backward residuals. These are multiplied by the coefficient matrices $K^b_{d,i}$ to produce the d-step predictor for $E\{y(t+d)|Y^t_{t-k+1}\}$ in the ladder part of the filter. Another part of the ladder filter, which is not strictly necessary if only predicted estimates of y are required and the $K^b_{d,i}$ are available, constructs the prediction error residuals of increasing order ; this part does not require any additional multiplication. The coefficient matrices $K^b_{d,i}$ are not provided by the lattice structure, but have to be either known or constructed.

D-step backward residuals of order k can be defined similarly.

$$g_{d,k}(t) \triangleq y\,(t-d-k+1) - E\{y(t-d-k+1)\,|\,Y^t_{t-k+1}\} \qquad (3.9)$$

$$g_{d,k+1}(t) = g_{d,k}(t-1) - K^f_{d,k+1}\, f_k(t) \qquad (3.10)$$

$$= y\,(t-d-k) - \sum_{i=0}^{k} K^f_{d,i+1}\, f_i(t-k+i) \qquad (3.11)$$

with $g_{d,0}(t) = y(t-d+1)$.

$$K^f_{d,i+1} = E\{y(t-d-i)f_i^T(t)\}(R_i^f)^{-1} \triangleq S^\times_{d,i}(R_i^f)^{-1}$$

Alternative expressions for $S^\times_{d,i}$ are :

$$S^\times_{d,i} = E\{y(t-d-i)f_i^T(t)\} = E\{g_{d,i}(t-1)f_i^T(t)\}$$

$$= E\{g_{d,i}(t-1)y^T(t)\} \qquad (3.12)$$

Finally, the d-step backward predictor for y(t) can be written as :

$$E\{y(t-d-k+1)\,|\,Y^t_{t-k+1}\} = \sum_{i=1}^{k} K^f_{d,i} f_{i-1}(t-k+i) \qquad (3.13)$$

In the last expression, we have used $\{f_0(t-k+1),\ f_1(t-k+2), \ldots,\ f_{k-1}(t)\}$ as a basis for Y^t_{t-k+1}. Note the complete symmetry between (3.7) and (3.10), (3.6) and (3.11), (3.5) and (3.12), (3.2) and (3.13). Finally note that, for d=1, the expressions of Section 3 become identical to those of Section 2 with $f_{1,k}(t) = f_k(t)$, $g_{1,k}(t) = g_k(t)$, $S_{1,k} = S_k = S^{\times T}_{1,k}$, $K^f_{1,k} = K^f_k$, and $K^b_{1,k} = K^b_k$.

Expressions (3.7) and (3.10) provide recursive relations for $f_{d,k}(t)$ and $b_{d,k}(t)$, where the recursion is on k, the order of the predictor.

IV THE RECURSIVE LEAST-SQUARES D-STEP LADDER FILTER

We present now the adaptive implementation of the filter of section 3 using a recursive least squares procedure.
For simplicity of notations we shall assume without loss of generality that the observation process is scalar. Given the data $\{y(j),\ 0 \leq j \leq t\}$ the one-step and d step forward and backward residuals of order k, defined in the previous sections, will have the following form :

$$f_k(t) = y(t) - \sum_{i=1}^{k} a_{k,i}(t)y(t-i) = A_k(t)^T\phi_k(t) \qquad (4.1)$$

$$g_k(t) = y(t-k) - \sum_{i=1}^{k} b_{k,i}(t)y(t-k+i) = B_k(t)^T\phi_k(t) \qquad (4.2)$$

$$f_{d,k}(t) = y(t) - \sum_{i=1}^{k} a_{d,k,i}(t)y(t-d-i+1) = A_{d,k}(t)^T\,\phi_{d+k-1}(t) \qquad (4.3)$$

$$g_{d,k}(t) = y(t-d-k+1) - \sum_{i=1}^{k} b_{d,k,i}(t)y(t-k+i) = B_{d,k}(t)^T\,\phi_{d+k-1}(t) \qquad (4.4)$$

where

$$A_k(t)^T = [\,1 - a_{k,1}(t) \ldots - a_{k,k}(t)]$$

$$B_k(t)^T = [\,-b_{k,k}(t) \quad -b_{k,1}(t)\ \ 1]$$

$$A_{d,k}(t)^T = [\,1\ \overbrace{0\ \ldots\ 0}^{d-1} - a_{d,k,1}(t) - a_{d,k,k}(t)]$$

$$B_{d,k}(t)^T = [\,-b_{d,k,k}(t)\ \ldots - b_{d,k,1}(t)\ \underbrace{0 - 0}_{d-1}\ 1]$$

$$\phi_k(t)^T = [\,y(t)\ y(t-1) \ldots\ y(t-k)]$$

We also define

$$Y_k(t) = \begin{bmatrix} y(0) & y(1) & \cdots & y(k) & \cdots & y(t) \\ 0 & y(0) & & & & \cdot \\ \cdot & \ddots & \ddots & & & \\ \cdot & & \ddots & \ddots & & \\ \cdot & & & \ddots & & \\ 0 & 0 & & y(0) & & y(t-k) \end{bmatrix} \qquad (4.5)$$

and the sample covariance matrix

$$R_k(t) = Y_k(t)\ Y_k(t)^T \qquad (4.6)$$

It will be assumed throughout that $R_k(t)$ is non-singular.
The coefficient vectors $A_k(t)$, $B_k(t)$, $A_{d,k}(t)$ and $B_{d,k}(t)$ are defined at each time as the solutions of the following minimization problems : min $A_k(t)^T R_k(t)\ A_k(t)$, min $B_k(t)^T R_k(t)\ B_k(t)$, min $A_{d,k}(t)^T\ R_{d+k-1}(t)\ A_{d,k}(t)$, min $B_{d,k}(t)^T\ R_{d+k-1}(t)\ B_{d,k}(t)$. This amounts to minimizing the sum of the squares of the residuals from 0 to t, but all evaluated with the same coefficient $A_k(t)$ (resp $B_k(t)$, $A_{d,k}(t)$, $B_{d,k}(t)$).
The minimization yields the following equations :

$$R_k(t)A_k(t) = \begin{bmatrix} R^f_k(t) \\ 0 \\ \cdot \\ \cdot \\ \cdot \\ 0 \end{bmatrix} \begin{matrix} \}1 \\ \\ \}k \\ \\ \\ \end{matrix}, \quad R_k(t)B_k(t) = \begin{bmatrix} 0 \\ \cdot \\ \cdot \\ \cdot \\ 0 \\ R^b_k(t) \end{bmatrix} \begin{matrix} \\ \\ \}k \\ \\ \\ \}1 \end{matrix} \qquad (4.7)$$

$$R_{d+k-1}(t)A_{d,k}(t) = \begin{bmatrix} R^f_{d,k}(t) \\ x \\ \cdot \\ \cdot \\ x \\ 0 \\ \cdot \\ \cdot \\ 0 \end{bmatrix} \begin{matrix} \}1 \\ \\ \}d-1 \\ \\ \\ \\ \}k \\ \\ \\ \end{matrix} \qquad (4.8a)$$

$$R_{d+k-1}(t)B_{d,k}(t) = \begin{bmatrix} 0 \\ \cdot \\ \cdot \\ 0 \\ x \\ \cdot \\ \cdot \\ x \\ R^{b}_{d,k}(t) \end{bmatrix} \begin{matrix} \left.\right\} k \\ \\ \left.\right\} d-1 \\ \\ 1 \end{matrix} \qquad (4.8b)$$

The last k equations on the left of (4.7) and of (4.8a) define $A_k(t)$ and $A_{d,k}(t)$, while the first equation defines $R^f_k(t)$ and $R^f_{d,k}(t)$ and similarly for $B_k(t)$ and $B_{d,k}(t)$. $R^f_k(t)$, $R^f_{d,k}(t)$, $R^b_k(t)$, $R^b_{d,k}(t)$ are the sample covariances of the optimal residuals.

The remainder of the section consists in finding recursive expressions for the coefficient vectors $A_k(t)$, $B_k(t)$, $A_{d,k}(t)$, $B_{d,k}(t)$ (recursions both in the order k and in the time t) without having to invert the matrix $R_k(t)$ or $R_{d+k-1}(t)$. These recursions will then lead to an implementation in lattice form. The recursive formulas are derived from the following recursions on $R_k(t)$.

$$R_k(t+1) = R_k(t) + \phi_k(t+1)\,\phi_k(t+1)^T \qquad (4.9)$$

$$R_{k+1}(t) = \begin{bmatrix} X & X \\ X & R_k(t-1) \end{bmatrix} = \begin{bmatrix} R_k(t) & X \\ X & X \end{bmatrix} \qquad (4.10)$$

$$R_{d+k}(t) = \begin{bmatrix} X & X \\ X & R_k(t-d) \end{bmatrix} = \begin{bmatrix} R_k(t) & X \\ X & X \end{bmatrix} \qquad (4.11)$$

where the X are submatrices of appropriate dimensions. We will also need two auxiliary quantities

a (k+1) - vector: $C_k(t) = R_k(t)^{-1}\,\phi_k(t)$ (4.12)

a scalar : $\gamma_k(t) = \phi_k(t)^T R_k(t)^{-1}\phi_k(t) = \phi_k(t)^T C_k(t)$

Note that $0 \leqslant \gamma_k(t) < 1$.

Order and time update recursions for the one step prediction filter have been derived in [4]-[5].(see also [10]). Hence, in order to keep this paper within reasonable limits, we will directly give the final formulas.

Order Update equations :

$$A_{k+1}(t) = \begin{bmatrix} A_k(t) \\ \\ 0 \end{bmatrix} - \begin{bmatrix} 0 \\ \\ B_k(t-1) \end{bmatrix} \frac{S_k(t)}{R^b_k(t-1)}$$

$$R^f_{k+1}(t) = R^f_k(t) - \frac{S_k^2(t)}{R^b_k(t-1)}$$

$$B_{k+1}(t) = \begin{bmatrix} 0 \\ \\ B_k(t-1) \end{bmatrix} - \begin{bmatrix} A_k(t) \\ \\ 0 \end{bmatrix} \frac{S_k(t)}{R^f_k(t)}$$

$$R^b_{k+1}(t) = R^b_k(t-1) - \frac{S_k^2(t)}{R^f_k(t)}$$

$$C_{k+1}(t) = \begin{bmatrix} C_k(t) \\ \\ 0 \end{bmatrix} + B_{k+1}(t)\,\frac{g_{k+1}(t)}{R^b_{k+1}(t)}$$

$$\gamma_{k+1}(t) = \gamma_k(t) + \frac{g^2_{k+1}(t)}{R^b_{k+1}(t)}$$

where $S_k(t)$ is defined as :

$$S_k(t) = [\text{last row of } R_{k+1}(t)] \begin{bmatrix} A_k(t) \\ \\ 0 \end{bmatrix}$$

Time Update equations :

$$A_k(t+1) = A_k(t) - \begin{bmatrix} 0 \\ \\ C_{k-1}(t) \end{bmatrix} A_k(t)^T \phi_k(t+1)$$

$$R^f_k(t+1) = R^f_k(t) + \frac{f_k^2(t+1)}{1-\gamma_{k-1}(t)}$$

$$B_k(t+1) = B_k(t) - \begin{bmatrix} C_{k-1}(t+1) \\ \\ 0 \end{bmatrix} B_k^T(t)\,\phi_k(t+1)$$

$$R^b_k(t+1) = R^b_k(t) + \frac{g_k^2(t+1)}{1-\gamma_{k-1}(t+1)}$$

$$C_k(t+1) = \begin{bmatrix} 0 \\ \\ C_{k-1}(t) \end{bmatrix} + A_k(t+1)\,\frac{f_k(t+1)}{R^f_k(t+1)}$$

$$\gamma_k(t+1) = \gamma_{k-1}(t) + \frac{f_k^2(t+1)}{R^f_k(t+1)}$$

$$S_{k+1}(t+1) = S_{k+1}(t) + \frac{g_k(t)\,f_k(t+1)}{1-\gamma_{k-1}(t)}$$

The time update recursions for $R^f_k(t)$, $R^b_k(t)$, $\gamma_k(t)$ and $S_k(t)$, together with the order recursions for $A_k(t)$ and $B_k(t)$ constitute a complete set of recursions that are required for the adaptive implementation of the least-squares lattice filter. Assuming that k_{max} is the maximum order to be considered for the lattice filter, and that the filter has been adapted up to time t, then, when y(t+1) is observed, the following quantities must be updated for k = 1, 2, ..., k_{max} : $f_k(t+1)$, $g_k(t+1)$, $S_{k-1}(t+1)$, $R^b_{k-1}(t+1)$, $R^f_{k-1}(t+1)$, $\gamma_{k-2}(t+1)$.

If k_{max} is the maximum order of the lattice, then the time recursions can only start after a time t_1 such that $R_{k_{max}}(t_1)$ is nonsingular i.e $t_1 \geqslant k$. The initial conditions for

$S_k(t)$, $R_k^b(t)$, $R_k^f(t)$, $\gamma_k(t)$ are then computed using the nonrecursive (in time) expressions of the beginning of this section.
We recall that this adaptive implementation of the lattice filter is a recursive least-squares solution, which therefore converges to the optimal steady-state lattice filter in the case of stationary statistics. The amount of computations is higher than the stochastic approximation implementation of [6] - [7], but notice that no matrix inversion is required.
Order update recursions for the d-step prediction filter can be obtained as follows :
We have (equation 4.8) :

$$R_{d+k}(t)\, A_{d,k+1}(t) = \begin{bmatrix} R^f_{d,k+1}(t) \\ x \\ \cdot \\ \cdot \\ x \\ 0 \\ \cdot \\ \cdot \\ 0 \end{bmatrix} \begin{matrix} 1 \\ \\ \Big\} d-1 \\ \\ \\ \Big\} k+1 \\ \\ \end{matrix} \qquad (4.12)$$

where the last k+1 equations determine $a_{d,k+1,1}(t) \ldots, a_{d,k+1,k+1}(t)$ and the first equation defines $R^f_{d,k+1}(t)$. Using (4.10) we also have :

$$R_{d+k}(t) \begin{bmatrix} A_{d,k}(t) \\ \\ 0 \end{bmatrix} = \left[\begin{array}{c|c} R_{d+k-1}(t) & X \\ \hline X & X \end{array}\right] \begin{bmatrix} A_{d,k}(t) \\ \\ 0 \end{bmatrix}$$

$$= \begin{bmatrix} R^f_{d,k}(t) \\ 0 \\ \cdot \\ 0 \\ S_{d,k}(t) \end{bmatrix} \qquad (4.13)$$

where $S_{d,k}(t)$ is defined as :

$$S_{d,k}(t) = [\text{last row of } R_{d+k}(t)] \begin{bmatrix} A_{d,k}(t) \\ \\ 0 \end{bmatrix}$$

One can write $A_{d,k+1}(t) = \begin{bmatrix} A_{d,k}(t) \\ \\ 0 \end{bmatrix} + \alpha \begin{bmatrix} 0 \\ \cdot \\ 0 \\ B_k(t-d) \end{bmatrix}$

and the problem is then to find the value of α.
In order to satisfy (4.12), α must be chosen such that :

$$S_{d,k}(t) + \alpha \;[\text{last row of } R_{d+k}(t)] \begin{bmatrix} 0 \\ \cdot \\ \cdot \\ 0 \\ B_k(t-d) \end{bmatrix} = 0$$

but, by equations (4.11) and (4.7), we have

$$[\text{last row of } R_{d+k}(t)] \begin{bmatrix} 0 \\ \cdot \\ \cdot \\ 0 \\ B_d(t-d) \end{bmatrix} = R^b_k(t-d)$$

Hence :

$$A_{d,k+1}(t) = \begin{bmatrix} A_{d,k}(t) \\ \\ 0 \end{bmatrix} - \begin{bmatrix} 0 \\ \cdot \\ \cdot \\ 0 \\ B_k(t-d) \end{bmatrix} \frac{S_{d,k}(t)}{R^b_k(t-d)} \qquad (4.14)$$

Premultiplying (4.14) by $R_{d+k}(t)$ and using (4.12) and (4.13) yields :

$$R^f_{d,k+1}(t) = R^f_{d,k}(t) - \frac{T^{\times}_{d,k}(t)\, S_{d,k}(t)}{R^b_k(t-d)}$$

Where $T^{\times}_{d,k}(t) = [\text{first row of } R_{d+k}(t)] \begin{bmatrix} 0 \\ \cdot \\ \cdot \\ 0 \\ B_k(t-d) \end{bmatrix}$

By the same argument, one finds the order update equations for $B_{d,k}(t)$:

$$B_{d,k+1}(t) = \begin{bmatrix} 0 \\ \\ B_{d,k}(t-1) \end{bmatrix} - d\left\{ \begin{bmatrix} A_k(t) \\ 0 \\ \cdot \\ 0 \end{bmatrix} \right. \frac{S^{\times}_{d,k}(t)}{R^f_k(t)}$$

$$R^b_{d,k+1}(t) = R^b_{d,k}(t) - \frac{T_{d,k}(t) \quad S^{\times}_{d,k}(t)}{R^f_k(t)}$$

where $S^{\times}_{d,k}(t) = [\text{first row of } R_{d+k}(t)] \begin{bmatrix} 0 \\ \\ B_{d,k}(t-l) \end{bmatrix}$

and $T_{d,k}(t) = [\text{last row of } R_{d+k}(t)] \begin{bmatrix} A_k(t) \\ 0 \\ \cdot \\ \cdot \\ 0 \end{bmatrix} \Big\} d$

We now derive time update recursions for $A_{d,k}(t)$, $B_{d,k}(t)$, $S_{d,k}(t)$ and $S^{\times}_{d,k}(t)$.
By (4.9) and (4.8)

$$R_{d+k-1}(t+1)\, A_{d,k}(t) = \begin{bmatrix} R^f_{d,k}(t) \\ x \\ \cdot \\ \cdot \\ x \\ 0 \\ \cdot \\ \cdot \\ 0 \end{bmatrix}$$

$$+\ \phi_{d+k-1}(t+1)\, \phi_{d+k-1}(t+1)^T A_{d,k}(t) \qquad (4.15)$$

and by (4.10)

$$R_{d+k-1}(t+1) \begin{bmatrix} 0 \\ \\ C_{d+k-2}(t) \end{bmatrix} = \left[\begin{array}{c|c} X & X \\ \hline X & R_{d+k-2}(t) \end{array}\right] x$$

$$\begin{bmatrix} 0 \\ \\ C_{d+k-2}(t) \end{bmatrix} = \begin{bmatrix} X \\ \\ \phi_{d+k-2}(t) \end{bmatrix} \qquad (4.16)$$

Using (4.8), (4.15) and (4.16), and remembering that $A_{d,k}(t+1)$ is defined by the last k equations of (4.8), we have :

$$A_{d,k}(t+1) = A_{d,k}(t) - \begin{bmatrix} 0 \\ \\ C_{d+k-2}(t) \end{bmatrix} \times \phi_{d+k-1}(t+1)^T A_{d,k}(t) \quad (4.17)$$

Premultiplying (4.17) by $R_{d+k-1}(t+1)$ and keeping only the first equation leads to :

$$R^f_{d,k}(t+1) = [\text{first row of } R_{d+k-1}]\ A_{d,k}(t) - [\text{first row of } R_{d+k-1}(t+1)] \begin{bmatrix} 0 \\ \\ C_{d+k-2}(t) \end{bmatrix} \times \phi_{d+k-1}(t+1)^T A_{d,k}(t).$$

The first term on the right hand side of the last equation yields, using (4.9) and (4.15) :

$$R^f_{d,k}(t) + y(t+1)\ \phi_{d+k-1}(t+1)^T A_{d,k}(t)$$

The second term yields, using (4.7), (4.12) and the time update recursion for $C_k(t)$:

$$[-y(t+1) + f_{d+k-1}(t+1)]\ \phi_{d+k-1}(t+1)^T A_{d,k}(t).$$

Combining these two terms we have a time update recursion for $R^f_{d,k}(.)$:

$$R^f_{d,k}(t+1) = R^f_{d,k}(t) + f_{d+k-1}(t)\ \phi_{d+k-1}(t+1)^T A_{d,k}(t).$$

Using the same procedures, we get time update recursions for $B_{d,k}(.)$ and $R^b_{d,k}(.)$:

$$B_{d,k}(t+1) = B_{d,k}(t) - \begin{bmatrix} C_{d+k-2}(t+1) \\ \\ 0 \end{bmatrix} \times \phi_{d+k-1}(t+1)^T B_{d,k}(t)$$

$$R^b_{d,k}(t+1) = R^b_{d,k}(t) + g_{d+k-1}(t+1)\ \phi_{d+k-1}(t+1)^T \times B_{d,k}(t).$$

Finally, we need to find time-update recursions for $S^{\times}_{d,k}(.)$ since these are needed in the order recursions for $A_{d,k}(t)$ and $B_{d,k}(t)$. Using the definition of $S_{d,k}(t+1)$ and the time update recursion (4.17) for $A_{d,k}(t+1)$, we have :

$$S_{d,k}(t+1) = [\text{last row of } R_{d+k}(t+1)] \times \left[\begin{bmatrix} A_{d,k}(t) \\ \\ 0 \end{bmatrix} - \begin{bmatrix} 0 \\ C_{d+k-2}(t) \\ 0 \end{bmatrix} \phi_{d+k-1}(t+1)^T A_{d,k}(t) \right]$$

The first term on the right hand side yields, using (4.9) and (4.12)

$$S_{d,k}(t) + y_{t-d-k+1}\ \phi_{d+k-1}(t+1)^T A_{d,k}(t)$$

The second term yields, using (4.10), (4.12) and the order recursion for $C_k(t)$:

$$- y_{t-d-k+1}\ \phi_{d+k-1}(t+1)^T A_{d,k}(t) + g_{d+k-1}(t)\ \phi_{d+k-1}(t+1)^T A_{d,k}(t)$$

Combining both terms gives a time update recursion for $S_{d,k}(.)$:

$$S_{d,k}(t+1) = S_{d,k}(t) + g_{d+k-1}(t)\ \phi_{d+k-1}(t+1)^T A_{d,k}(t).$$

Using a similar procedure, yields a time update recursion for $S^{\times}_{d,k}(.)$:

$$S^{\times}_{d,k}(t+1) = S^{\times}_{d,k}(t) + f_{d+k-1}(t+1)\ \phi_{d+k-1}(t+1)^T B_{d,k}(t).$$

The time update recursions for $A_{d,k}(t)$, $B_{d,k}(t)$, $S_{d,k}(t)$, $S^{\times}_{d,k}(t)$ together with the order update recursions for $A_{d,k}(t)$ and $B_{d,k}(t)$ and the recursions of the basic one-step prediction filter constitute a complete set of recursions for the least squares computation of the d-step residuals.

Notice that the recursions on $R^f_{d,k}(t)$ and $R^b_{d,k}(t)$ are only needed if the sum of squares of these residuals is desired.

V THE D-STEP AHEAD PREDICTOR.

We show now that, with a slight modification, the recursive d-step ladder filter of section IV can be transformed into a d-step ahead prediction filter.

We have seen in the previous sections that $f_{d,k}(t)$ could be expressed in two different ways.

$$f_{d,k}(t) = y(t) - \sum_{i=1}^{k} K^b_{d,i}(t)\ g_{i-1}(t-d)$$

$$\text{where } K^b_{d,i+1}(t) = \frac{S^b_{d,i}(t)}{R^b_i(t-d)}$$

or

$$f_{d,k}(t) = y(t) - \sum_{i=1}^{k} a_{d,k,i}(t)\ y(t-d-i+1)$$

$$\text{Hence } \sum_{i=1}^{k} K^b_{d,i}(t)\ g_{i-1}(t-d) = \sum_{i=1}^{k} a_{d,k,i}(t)\ y(t-d-i+1) \quad (5.1)$$

Any one of the two expressions could be used to define the least squares predictor of $y(t)$ based on $\{y(t-d)\ldots y(t-d-k+1)\}$ except that the coefficients $a_{d,k,i}(t)$ and $K^b_{d,i}(t)$ depend on all observations up to time t, which makes theses expressions non causal.

A truly causal least squares d-step ahead predictor could be obtained by replacing the coefficients $a_{d,k,i}(t)$ (resp. $K^b_{d,i}(t)$) in (5.1) by $a_{d,k,i}(t-d)$ (resp. $K^b_{d,i}(t-d)$).

We shall define the d-step ahead predictor as :

$$\hat{y}_{d,k}(t) = \sum_{i=1}^{k} K^{b}_{d,i}(t-d)\ g_{i-1}(t-d) \qquad (5.2)$$

It turns out that this predictor is different from one that would have been defined as :

$$\tilde{y}_{d,k}(t) = \sum_{i=1}^{k} a_{d,k,i}(t-d)\ y(t-d-i+1) \qquad (5.3)$$

In fact, it can be shown that expression (5.3) is equivalent with

$$\tilde{y}_{d,k}(t) = \sum_{i=1}^{k} K^{b'}_{d,i}(t-d)\ \tilde{g}_{i-1}(t-d)$$

in which $\tilde{g}_{i-1}(t-d)$ are estimates of $g_{i-1}(t-d)$ obtained with coefficients computed at time t-2d :

$$\tilde{g}_{i-1}(t-d) = \phi_{i-1}(t-d)^{T}\ B_{i-1}(t-2d)$$

Hence, we chose expression (5.2) which uses the most recent estimates of the coefficients $K^{b}_{d,i}(t-d)$ and the backward residuals $g_{i-1}(t-d)$. As can be seen from Fig.3, the d-step ahead prediction is directly obtained from the basic lattice filter.

VI CONCLUSION.

We have shown that the orthogonalizing property of the basic linear prediction lattice filter can be used to construct a d-step predictor filter in lattice and ladder form.

We have derived the equations for the d-step predictions and prediction errors assuming known covariances first.

We have then given an adaptive implementation using least squares recursions. The d-step ladder filter is obtained from the basic recursive least squares lattice filter with few additional computations.

In addition, we have shown how the d-step ladder filter can be used to generate d-step ahead recursive least squares predictions for the observation process.

REFERENCES.

[1.] F. ITAKURA, S. SAITO, "Digital Filtering Techniques for Speech Analysis and Synthesis", 7th Int. Congress Acoustics, Paper 25 C-1, Budapest, 1971.

[2.] N. LEVINSON, "The Wiener RMS Error Criterion in Filter Design and Prediction", J.Math.Phys., Vol.25, pp.261-278, 1947.

[3.] A. LINDQUIST, "A new Algorithm for Optimal Filtering of Discrete Time Stationary Processes", SIAM J. of control, Vol.12, N° 4 pp.736-746, Nov.1974.

[4.] M. MORF, A. VIERA, D.T. LEE, "Ladder Forms for Identification and Speech Processing", IEEE Conf. on Decision and Control, pp.1074-1078, 1977.

[5.] M. MORF, D.T. LEE, "Recursive Least-Squares Ladder Forms for Fast Parameter Tracking", IEEE Conf. on Decision and Control, pp.1362-1367, 1978.

[6.] L.J. GRIFFITHS, "A Continously Adaptative Filter Implemented as a Lattice Structure", IEEE Int. Conf. Acoust., Speech, Signal Processing, Hartford, pp.583-586, May 1977.

[7.] J. MAKHOUL, "A Class of All-zero Lattice Digital Filters : Properties and Applications", IEEE Trans. Acoustics, Speech, Signal Processing, Vol. 26, pp.304-314, Aug. 1978.

[8.] A.H. GRAY, J. MARKEL, "Digital Lattice and Ladder Filter Synthesis", IEEE Trans. Audio and Electroacoustics, Vol.21, pp.491-500, Dec.1973.

[9.] M. GEVERS, V. WERTZ, "A D-Step Predictor in Lattice and Ladder Form", Oct. 1980, submitted for publication.

[10] M. GEVERS, V.WERTZ, "A Recursive Least-Squares D-Step ahead Predictor in Ladder Form", Dec. 1980, submitted for publication.

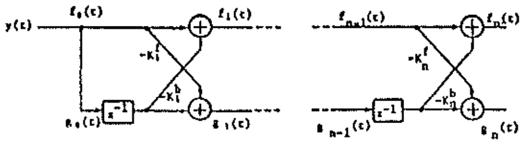

Figure 1

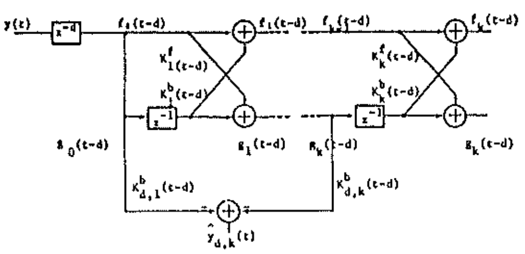

Figure 3.

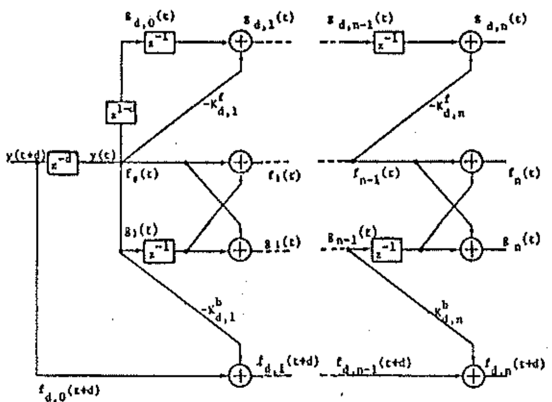

Figure 2.