A STATE-SPACE APPROACH TO ADAPTIVE FILTERING *

ALI H. SAYED AND THOMAS KAILATH

Information Systems Laboratory, Stanford University, Stanford, CA 94305, USA.

ABSTRACT

We describe a unified square-root-based derivation of adaptive filtering schemes that is based on reformulating the original problem as a state-space linear least-squares estimation problem. In this process we encounter rich connections with algorithms that have been long established in linear least-squares estimation theory such as the Kalman filter, the Chandrasekhar filter, and the information forms of the Kalman and Chandrasekhar algorithms. The approach also suggests some generalizations and extensions of classical results.

1. INTRODUCTION

Adaptive filtering is widely used to cope with time-variations of system parameters and to compensate for the lack of a priori knowledge in the statistical properties of the input data. For a variety of reasons, the least–squares criterion is used to derive linear estimators for the desired parameters, and a wide range of algorithms and schemes has been developed that exploits the data structure and reduces the computational complexity. These basically fall into four main groups of adaptive filtering algorithms (see, *e.g.*,[1, 2, 3] and the references therein for extensive discussions): **R**ecursive **L**east **S**quares (RLS) and the corresponding fast versions, **L**attice **L**east–**S**quares (LLS), **QR**-based least–squares, and gradient-based algorithms. The different adaptive schemes have been derived in a variety of methods; the derivations are usually lengthy and the resulting algorithms are normally described by a large set of equations and variables. Moreover, the connections between the different algorithms are usually obscured by the different derivations. Our purpose is to describe an alternative approach, which is unifying and clarifies the connections that exist among the different algorithms. We pose the original adaptive problem as a Kalman filtering problem, which has been well studied and investigated since the early sixties. We then show that the different adaptive algorithms can be interpreted as old and long established algorithms in state-space linear estimation theory. Moreover, this approach already suggests some generalizations and extensions of classical results [4, 5, 6, 7].

* This work was supported in part by the Air Force Office of Scientific Research, Air Force Systems Command under Contract AFOSR91-0060. The work of the first author was also supported by a fellowship from Fundação de Amparo à Pesquisa do Estado de São Paulo and by Escola Politécnica da Universidade de São Paulo, Brazil.

2. STATE-SPACE ESTIMATION ALGORITHMS

We first review some basic algorithms in linear least-squares (lls) estimation. We consider a $p \times 1$ process $\{\mathbf{y}_i\}$ with an n-dimensional state-space model of the form

$$\mathbf{x}_{i+1} = F_i\mathbf{x}_i\ , \quad \mathbf{y}_i = H_i\mathbf{x}_i + \mathbf{v}_i \qquad (1)$$

where F_i and H_i are $n \times n$ and $p \times n$ known matrices, respectively. We assume that $\mathbf{x}_0$ and $\mathbf{v}_i$ are uncorrelated stochastic variables with $E\mathbf{x}_0 = \bar{\mathbf{x}}_0, E\mathbf{v}_i = \mathbf{0}$, $E(\mathbf{x}_0 - \bar{\mathbf{x}}_0)(\mathbf{x}_0 - \bar{\mathbf{x}}_0)^* = \Pi_0$, and $E\mathbf{v}_i\mathbf{v}_j^* = R_i\delta_{ij}$. Let $\hat{\mathbf{x}}_{i|i-1}$ and $\hat{\mathbf{y}}_{i|i-1}$ denote the linear least-squares estimates of $\mathbf{x}_i$ and $\mathbf{y}_i$ given the past i observations $\{\mathbf{y}_0, \ldots, \mathbf{y}_{i-1}\}$, respectively. The recursive procedure that relates $\hat{\mathbf{x}}_{i+1|i}$ to $\hat{\mathbf{x}}_{i|i-1}$ is a *special case* of the well-known Kalman filter algorithm, whose derivation is straightforward once we invoke the innovations concept. Define the innovations process $\epsilon_i = \mathbf{y}_i - \hat{\mathbf{y}}_{i|i-1}$, which represents the new information in $\mathbf{y}_i$ that is not contained in the space spanned by the past observations. The point is that the innovations are uncorrelated with each other ($E\epsilon_i\epsilon_j^* = R_{\epsilon,i}\delta_{ij}$), which readily implies that $\hat{\mathbf{x}}_{i+1|i} = F_i\hat{\mathbf{x}}_{i|i-1} + K_iR_{\epsilon,i}^{-1}\epsilon_i$, where we defined $K_i = E\mathbf{x}_{i+1}\epsilon_i^*$. If we introduce the covariance matrix $P_{i|i-1} = E(\mathbf{x}_i - \hat{\mathbf{x}}_{i|i-1})(\mathbf{x}_i - \hat{\mathbf{x}}_{i|i-1})^*$, it then follows that $R_{\epsilon,i} = H_iP_{i|i-1}H_i^* + R_i$ and $K_i = F_iP_{i|i-1}H_i^*$. It also follows that $P_{i|i-1}$ satisfies the following Riccati difference recursion: $P_{0|-1} = \Pi_0$, $P_{i+1|i} = F_iP_{i|i-1}F_i^* - K_iR_{\epsilon,i}^{-1}K_i^*$.

The Kalman filter propagates the Riccati variable $P_{i|i-1}$. In some applications, however, the initial uncertainty in $\mathbf{x}_0$ can be high (that is, $\Pi_0 \to \infty$). For these cases it is preferable to propagate $P_{i|i-1}^{-1}$, and the resulting algorithm is usually referred to as the Information filter [8], which can be easily verified to be given by the equations

$$\begin{aligned} P_{i+1|i}^{-1} &= F_i^{-*}\left[P_{i|i-1}^{-1} + H_i^*R_i^{-1}H_i\right]F_i^{-1} \\ R_{\epsilon,i}^{-1} &= R_i^{-1} - R_i^{-1}H_iF_i^{-1}P_{i+1|i}F_i^{-*}H_i^*R_i^{-1} \\ P_{i+1|i}^{-1}\hat{\mathbf{x}}_{i+1|i} &= F_i^{-*}\left[P_{i|i-1}^{-1}\hat{\mathbf{x}}_{i|i-1} + H_i^*R_i^{-1}\mathbf{y}_i\right] \end{aligned}$$

0-7803-0946-4/93 $3.00 © 1993 IEEE

It is also straightforward to verify that these equations can be grouped together and rewritten in square-root form as follows (see, *e.g.*, [8]):

$$\begin{bmatrix} F_i^{-*}P_{i|i-1}^{-*/2} & F_i^{-*}H_i^*R_i^{-*/2} \\ \hat{\mathbf{x}}_{i|i-1}^*P_{i|i-1}^{-*/2} & \mathbf{y}_i^*R_i^{-*/2} \\ \mathbf{0} & R_i^{-*/2} \end{bmatrix} \Theta_i = \begin{bmatrix} P_{i+1|i}^{-*/2} & \mathbf{0} \\ \hat{\mathbf{x}}_{i+1|i}^*P_{i+1|i}^{-*/2} & \epsilon_i^*R_{\epsilon,i}^{-*/2} \\ R_i^{-1}H_iF_i^{-1}P_{i+1|i}^{1/2} & R_{\epsilon,i}^{-*/2} \end{bmatrix} \tag{2}$$

where Θ_i is any unitary matrix that produces the zero entry in the postarray.

The number of operations needed in going from index i to index $(i+1)$ in either the Riccati-based filter or the information filter is $O(n^3)$, and this is true whether or not the state-space model has constant parameters. However, one expects a computationally more efficient procedure in the case of time-variant models that exhibit certain structure in their time-variation. For this purpose, the state-space model (1) will be said to be *structured* if there exist $n \times n$ matrices Ψ_i such that F_i and H_i vary according to the following rules: $H_i = H_{i+1}\Psi_i$ and $F_{i+1}\Psi_i = \Psi_{i+1}F_i$. We further assume that the covariance matrices R_i are constant for all i ($R_i = R$). Define $\bar{K}_{p,i} = K_iR_{\epsilon,i}^{-*/2}$, and let S be an $\alpha \times \alpha$ signature matrix defined as follows: $\alpha =$ rank $(P_{1|0} - \Psi_0P_{0|-1}\Psi_0^* \equiv L_0SL_0^*)$. It can be easily shown that for such models the quantities $\bar{K}_{p,i}$ and $R_{\epsilon,i}$ can be alternatively propagated via the so-called extended Chandrasekhar equations [5, 9, 10]:

$$\begin{bmatrix} R_{\epsilon,i}^{1/2} & H_{i+1}L_i \\ \Psi_{i+1}\bar{K}_{p,i} & F_{i+1}L_i \end{bmatrix} \Theta_i = \begin{bmatrix} R_{\epsilon,i+1}^{1/2} & \mathbf{0} \\ \bar{K}_{p,i+1} & L_{i+1} \end{bmatrix} \tag{3}$$

where Θ_i is any $(I \oplus S)-$unitary matrix that produces the zero entry in the postarray. For sparse enough matrices Ψ_i, the number of operations needed per iteration is $O(n^2)$ if $p \ll n$ and $\alpha \ll n$.

3. THE RLS PROBLEM

The basic problem in linear least-squares estimation reads as follows: given pairs of data points, $\{\mathbf{u}_i, d(i)\}_{i=0}^N$, where $\mathbf{u}_i$ is a $1 \times M$ row vector that consists of the values of M input channels at time i, $\mathbf{u}_i = [\ u_1(i) \ \ldots \ u_M(i)\ ]$ ($d(i)$ and $u_j(i)$ are assumed scalar for simplicity), we are required to determine the linear least-squares estimate (denoted by $\mathbf{w}_N$) of an $M \times 1$ column vector of unknown tap weights $\mathbf{w}$ so as to minimize the exponentially weighted sum $\sum_{i=0}^N \lambda^{N-i}|d(i) - \mathbf{u}_i\mathbf{w}|^2$, $0 < \lambda \leq 1$, which is equivalent to minimizing over $\mathbf{x_0} = \mathbf{w}$ the expression $\sum_{i=0}^N |y(i) - \mathbf{u}_i\mathbf{x}_i|^2$, where we defined the normalized quantities $y(i) = d(i)/(\sqrt{\lambda})^i$ and $\mathbf{x}_i = \mathbf{w}/(\sqrt{\lambda})^i$. Now, the latter expression can be easily recast into a Kalman filtering problem by considering the following $M-$dimensional model

$$\begin{aligned} \mathbf{x}_{i+1} &= \lambda^{-1/2}\mathbf{x}_i, \quad \mathbf{x}_0 = \mathbf{w}, \quad \Pi_0 = \infty I \\ y(i) &= \mathbf{u}_i\mathbf{x}_i + v(i), \quad Ev(i)v^*(j) = \delta_{ij} \end{aligned} \tag{4}$$

Let $\mathbf{w}_i$ denote the weight vector estimate given the input data up to time i. This is clearly related to the state-estimate via $\mathbf{w}_i/(\sqrt{\lambda})^{i+1} = \hat{\mathbf{x}}_{i+1|i}$, which is given by $\hat{\mathbf{x}}_{0|-1} = \mathbf{0}$, $\hat{\mathbf{x}}_{i+1|i} = \lambda^{-1/2}\hat{\mathbf{x}}_{i|i-1} + \mathbf{k}_ir_{\epsilon,i}^{-1}\epsilon(i)$, where $\epsilon(i) = y(i) - \mathbf{u}_i\hat{\mathbf{x}}_{i|i-1}$, $\mathbf{k}_i = \lambda^{-1/2}P_{i|i-1}\mathbf{u}_i^*$, $r_{\epsilon,i} = 1 + \mathbf{u}_iP_{i|i-1}\mathbf{u}_i^*$, and $P_{i|i-1}$ is the error covariance matrix that satisfies the Riccati difference equation $P_{i+1|i} = \lambda^{-1}\left[P_{i|i-1} - P_{i|i-1}\mathbf{u}_i^*r_{\epsilon,i}^{-1}\mathbf{u}_iP_{i|i-1}\right]$. The Kalman filter variables $\{\mathbf{k}_i, r_{\epsilon,i}, P_{i+1|i}\}$ are scaled versions of the RLS variables as usually described in the literature. To clarify this point, we first note that (using $P_{0|-1}^{-1} = \mathbf{0}$) $P_{i+1|i}^{-1} = \lambda P_{i|i-1}^{-1} + \lambda\mathbf{u}_i^*\mathbf{u}_i = \lambda\sum_{j=0}^i \lambda^{i-j}\mathbf{u}_j^*\mathbf{u}_j \equiv \lambda\Phi_i$, where Φ_i is known as the weighted autocorrelation matrix. If we define the RLS variable $\mathbf{P}_i = \Phi_i^{-1}$, then we get $\mathbf{P}_i = \lambda P_{i+1|i}$. Moreover, the *a priori* error $e(i) = d(i) - \mathbf{u}_i\mathbf{w}_{i-1}$, is clearly related to the innovations by $\epsilon(i) = e(i)/(\sqrt{\lambda})^i$. The *a posteriori* error $e^p(i) = d(i) - \mathbf{u}_i\mathbf{w}_i$, can be written as $e^p(i) = d(i) - (\sqrt{\lambda})^{i+1}\mathbf{u}_i\hat{\mathbf{x}}_{i+1|i} = e(i)r_{\epsilon,i}^{-1}$. That is, the conversion factor γ_i, which converts the a priori error $e(i)$ to the a posteriori error $e^p(i)$, is given by $\gamma_i = r_{\epsilon,i}^{-1}$. We can finally, rewrite the Kalman filter equations in terms of the original RLS variables:

$$\begin{aligned} \mathbf{w}_i &= \mathbf{w}_{i-1} + \mathbf{g}_ie(i), \quad \mathbf{w}_{-1} = \mathbf{0} \\ \mathbf{g}_i &= \frac{\lambda^{-1}\mathbf{P}_{i-1}\mathbf{u}_i^*}{1 + \lambda^{-1}\mathbf{u}_i\mathbf{P}_{i-1}\mathbf{u}_i^*} \\ \mathbf{P}_i &= \lambda^{-1}\left[\mathbf{P}_{i-1} - \mathbf{g}_i\mathbf{u}_i\mathbf{P}_{i-1}\right], \quad \mathbf{P}_{-1} = \sigma I \end{aligned} \tag{5}$$

for $\sigma \gg 1$, and where we defined $\mathbf{g}_i = \sqrt{\lambda}\mathbf{k}_ir_{\epsilon,i}^{-1}$. In summary, the Kalman variables and the RLS variables are related as summarized in the following table.

KF	RLS
$y(i)$	$d(i)/(\sqrt{\lambda})^i$
$\mathbf{x}_i$	$\mathbf{w}/(\sqrt{\lambda})^i$
$\hat{\mathbf{x}}_{i+1\|i}$	$\mathbf{w}_i/(\sqrt{\lambda})^{i+1}$
$\lambda P_{i+1\|i}$	$\mathbf{P}_i = \Phi_i^{-1}$
$\sqrt{\lambda}\mathbf{k}_ir_{\epsilon,i}^{-1}$	$\mathbf{g}_i$
$\epsilon(i)$	$e(i)/(\sqrt{\lambda})^i$
$r_{\epsilon,i}^{-1}$	γ_i

For a variety of reasons, it is sometimes preferable to start the recursive algorithm with a *nonzero* initial value $\mathbf{w}_{-1}$. This change can be incorporated into our derivation by assuming $E\mathbf{x_0} = E\mathbf{w} = \bar{\mathbf{w}}$. This corresponds to the error criterion $\min_{\mathbf{w}}\left[(\mathbf{w} - \bar{\mathbf{w}})^*\Pi_0^{-1}(\mathbf{w} - \bar{\mathbf{w}}) + \sum_{i=0}^N \lambda^{N-i}|d(i) - \mathbf{u}_i\mathbf{w}|^2\right]$, with $E(\mathbf{w} - \bar{\mathbf{w}})(\mathbf{w} - \bar{\mathbf{w}})^* = \Pi_0$. The state-space model (4) remains unchanged except for the new values of $P_{0|-1} = \Pi_0$ and $\hat{\mathbf{x}}_{0|-1} = \bar{\mathbf{w}}$.

4. FAST RECURSIVE LEAST SQUARES

We now further assume that the input channels exhibit shift structure, viz., $u_j(i) = u_{j-1}(i-1)$, or $\mathbf{u}_i = [\ u(i) \quad u(i-1) \quad \ldots \quad u(i-M+1)\ ]$. To exploit this fact, we consider the following $(N+1)-$ (not $M-$) dimensional *time-variant* model

$$\begin{aligned} \mathbf{x}_{i+1} &= \lambda^{-1/2}\mathbf{x}_i, \quad \mathbf{x}_0 = [\ \mathbf{w}^{\mathbf{T}} \quad 0\ ]^{\mathbf{T}} \\ y(i) &= \mathbf{h}_i\mathbf{x}_i + v(i), \quad Ev(i)v^*(j) = \delta_{ij} \end{aligned} \tag{6}$$

where $\mathbf{x}_i$ is now an $(N+1)\times 1$ state-vector with trailing zeros and $\mathbf{h}_i = [\ u(i) \quad u(i-1) \quad \ldots \quad u(0) \quad 0_{N-i}\ ]$ is a $1\times(N+1)$ row vector. An initial state covariance matrix (with trailing zeros) is also assumed, viz., $E(\mathbf{x}_0 - \bar{\mathbf{x}}_0)(\mathbf{x}_0 - \bar{\mathbf{x}}_0)^* \equiv \Pi_0 \oplus 0 = P_{0|-1}$. The Kalman gain $\bar{\mathbf{k}}_{p,i} = \mathbf{k}_i r_{e,i}^{-*/2}$ also has trailing zeros, $\bar{\mathbf{k}}_{p,i}^* \equiv [\ \mathbf{c}_i^* \quad 0\ ]$, say. The computational complexity of the RLS algorithm is $O(M^2)$ operations (multiplications and additions) per time step. However, *though time-variant*, the special structure of $\mathbf{h}_i$, viz., $\mathbf{h}_i = \mathbf{h}_{i+1}Z$, can be further exploited to reduce the operation count to $O(M)$, where Z denotes the shift matrix with ones on the first subdiagonal. Observe that this relation (along with $F_{i+1}Z = ZF_i$, since $F_i = \lambda^{-1/2}I$) shows that the state-space model (6) is indeed structured. The reduction in operation count can now be achieved by using the extended Chandrasekhar recursions with $\Psi_i = Z$. So let $L_0 S L_0^* = P_{1|0} - ZP_{0|-1}Z^*$, where L_0 is clearly of the form $L_0^* = [\ \tilde{L}_0^* \quad 0\ ]$ and $\tilde{L}_0$ is $(M+1)\times\alpha$. Let $\tilde{\mathbf{h}}_i$ be the row vector of the first $M+1$ coefficients of $\mathbf{h}_i$. Then

$$\begin{bmatrix} r_{e,i}^{1/2} & \tilde{\mathbf{h}}_{i+1}\tilde{L}_i \\ \begin{bmatrix} 0 \\ \mathbf{c}_i \end{bmatrix} & \lambda^{-1/2}\tilde{L}_i \end{bmatrix} \Theta_i = \begin{bmatrix} r_{e,i+1}^{1/2} & 0 \\ \begin{bmatrix} \mathbf{c}_{i+1} \\ 0 \end{bmatrix} & \tilde{L}_{i+1} \end{bmatrix}$$

where Θ_i is *any* $(1\oplus S)-$unitary matrix that produces the zero entry on the right hand-side of the above expression. The computational complexity of each step is $O(\alpha M)$, where the value of α depends on the choice of Π_0. This recursion is a square-root version of the fast RLS algorithms (FAEST,FTF) discussed in the literature [11, 12]. We further remark that the connection between the Chandrasekhar recursions and fast RLS algorithms was also discussed in [13, 14] by considering a particular *time-invariant* state-space model. Our derivation addresses the same problem within the more general framework of *structured time-variant* models. This allows us to consider more general cases that arise for instance in multidimensional, multichannel, and nonlinear adaptive problems, as we show in [6]. Moreover, though we assumed that the channel inputs obey a shift structure, our approach makes it clear that we can also obtain fast algorithms for other cases as well [7]: for example, if the input vectors $\mathbf{u}_i$ satisfy a relation of the form $\mathbf{u}_i = \mathbf{u}_{i+1}\Psi$, for some constant matrix Ψ, then the state-space model (4) is still structured, and we can write down the corresponding extended Chandrasekhar recursions. Finally, our framework also allows us to go beyond the RLS approach and to derive the so-called QR and lattice algorithms, as we shall now show.

5. THE QR ALGORITHM

We can alternatively apply the square-root *information filter* (2) to the state-space model (4). This directly yields the well-known QR algorithm [2, 3] (by using the correspondence established in the previous table), viz.,

$$\begin{bmatrix} \sqrt{\lambda}\Phi_{i-1}^{1/2} & \mathbf{u}_i^* \\ \sqrt{\lambda}\alpha_{i-1}^* & d^*(i) \\ 0 & 1 \end{bmatrix} \Theta_i = \begin{bmatrix} \Phi_i^{1/2} & 0 \\ \alpha_i^* & \bar{e}^*(i) \\ \mathbf{u}_i\Phi_i^{-*/2} & \gamma_i^{1/2} \end{bmatrix}$$

where we introduced the (angle) normalized a posteriori error $\bar{e}(i) = e^*(i)\gamma_i^{1/2} = e^{*p}(i)\gamma_i^{-1/2}$, and defined $\alpha_i = \Phi_i^{-1/2}\theta_i$ with $\theta_i = \sum_{j=0}^{i}\lambda^{i-j}d(i)\mathbf{u}_j^*$. If we closely examine the QR array equation then it is clear that it can be interpreted as follows: if we want to (recursively) produce the new information that is in $d(i)$ and not in $\mathbf{u}_i$, then we form a pre-array as above and (orthogonally) triangularize it to obtain the (normalized) estimation error. This observation is very helpful in the derivation of the order-recursive lattice algorithm, as discussed below.

6. ORDER-RECURSIVE LS ALGORITHMS

Assume again that the input channels exhibit shift structure. We now describe an alternative so-called *lattice* filter implementation that leads to an order-update procedure as we further explain: let f_i^M denote the M^{th} order *a-posteriori* forward prediction error at time i, which is due to estimating $u(i)$ from the M input values $\{u(i-M),\ldots,u(i-1)\}$. Knowing f_i^M we shall try to determine f_i^{M+1}, which is the $(M+1)^{th}$ forward prediction error at time i in estimating $u(i)$ from the $M+1$ data points $\{u(i-M-1),\ldots,u(i-1)\}$. Notice that we are basically using the same input data as before except for $u(i-M-1)$, and it is clear that all we need to do is to update f_i^M in order to incorporate the new information that is in $u(i-M-1)$ and not in the previous data. This suggests that we introduce b_{i-1}^M, which is the M^{th} order *a posteriori* backward prediction error at time $i-1$ in estimating $u(i-M-1)$ from these M input values $\{u(i-M),\ldots,u(i-1)\}$. In other words, the new information that is required to update f_i^M to f_i^{M+1} is contained in b_{i-1}^M, so that we are reduced to the setting described at the end of the previous section. That is, the associated array equation (in terms of normalized quantities) is given by

$$\begin{bmatrix} \sqrt{\lambda}\bar{\Phi}_{M,i-2}^{b/2} & \bar{b}_{i-1}^{*M} \\ \sqrt{\lambda}\bar{\alpha}_{M,i-2}^{*b} & \bar{f}_i^{*M} \\ 0 & \bar{\gamma}_{i-1,M}^{1/2} \end{bmatrix} \bar{\Theta}_{M,i-1}^{b} = \begin{bmatrix} \bar{\Phi}_{M,i-1}^{b/2} & 0 \\ \bar{\alpha}_{M,i-1}^{*b} & \bar{f}_i^{*(M+1)} \\ b_{i-1}^{M}\bar{\Phi}_{M,i-1}^{-*b/2} & \bar{\gamma}_{i-1,M+1}^{1/2} \end{bmatrix}$$

where the normalized quantities are defined by $\bar{\Phi}^b_{M,i} = \sum_{j=0}^{i} \lambda^{i-j} \bar{b}^{*M}_j \bar{b}^M_j$, $\bar{\theta}^b_{M,i} = \sum_{j=0}^{i} \lambda^{i-j} \bar{b}^{*M}_j \bar{f}^M_{j+1}$, $\bar{\alpha}^b_{M,i} = \bar{\theta}^b_{M,i} / \bar{\Phi}^{b/2}_{M,i}$.

We are still left with the problem of getting an update recursion for the backward error. But this can be obtained by following a similar argument, leading to

$$\begin{bmatrix} \sqrt{\lambda}\bar{\Phi}^{f/2}_{M,i-1} & \bar{f}^{*M}_i \\ \sqrt{\lambda}\bar{\alpha}^{*f}_{M,i-1} & \bar{b}^{*M}_{i-1} \\ 0 & \bar{\gamma}^{1/2}_{i-1,M} \end{bmatrix} \bar{\Theta}^f_{M,i} = \begin{bmatrix} \bar{\Phi}^{f/2}_{M,i} & 0 \\ \bar{\alpha}^{*f}_{M,i} & \bar{b}^{*(M+1)}_i \\ f^M_i \bar{\Phi}^{-*f/2}_{M,i} & \bar{\gamma}^{1/2}_{i,M+1} \end{bmatrix}$$

where the normalized quantities are given by $\bar{\Phi}^f_{M,i} = \sum_{j=0}^{i} \lambda^{i-j} \bar{f}^{*M}_j \bar{f}^M_j$, $\bar{\theta}^f_{M,i} = \sum_{j=0}^{i} \lambda^{i-j} \bar{f}^{*M}_j \bar{b}^M_{j-1}$, $\bar{\alpha}^f_{M,i} = \bar{\theta}^f_{M,i} / \bar{\Phi}^{f/2}_{M,i}$. The above arrays constitute the *square-root* form of the so-called (order-recursive) QR least-squares lattice algorithm [2, 3], which is usually presented in an extensive set of equations. The above derivation yields the arrays directly by noting that the forward and backward prediction problems are special cases of the standard least-squares problem and by writing down the associated *information filters*.

7. LEAST-SQUARES LATTICE

The least-squares lattice filter now follows directly by squaring and comparing terms on both sides of the square-root arrays, leading to [2, 3]

$$\begin{aligned} b^{M+1}_i &= b^M_{i-1} - k^M_{b,i} f^M_i \\ f^{M+1}_i &= f^M_i - k^M_{f,i} b^M_{i-1} \end{aligned}$$

where $k^M_{f,i} = \bar{\theta}^b_{M,i-1} / \bar{\Phi}^b_{M,i-1}$, $k^M_{b,i} = \bar{\theta}^f_{M,i} / \bar{\Phi}^f_{M,i}$, and

$$\bar{\Phi}^f_{M,i} = \lambda \bar{\Phi}^f_{M,i-1} + \frac{|f^M_i|^2}{\bar{\gamma}_{i-1,M}}, \quad \bar{\Phi}^b_{M,i-1} = \lambda \bar{\Phi}^b_{M,i-2} + \frac{|b^M_{i-1}|^2}{\bar{\gamma}_{i-1,M}}$$

$$\bar{\theta}^f_{M,i} = \lambda \bar{\theta}^f_{M,i-1} + \frac{f^{*M}_i b^M_{i-1}}{\bar{\gamma}_{i-1,M}}, \quad \bar{\theta}^b_{M,i-1} = \lambda \bar{\theta}^b_{M,i-2} + \frac{f^M_i b^{*M}_{i-1}}{\bar{\gamma}_{i-1,M}}$$

8. JOINT-PROCESS ESTIMATION

We can also follow a similar argument as in the case of the order-update recursions and derive an order-recursive procedure for the llse of the input signal $d(i)$. This follows easily by considering the sequence of orthogonal backward errors, $\{\ldots, b^M_{i-1}, b^{M+1}_i, b^{M+2}_{i+1}, \ldots\}$, and using the following array equations

$$\begin{bmatrix} \sqrt{\lambda}\bar{\Phi}^{b/2}_{M,i-2} & \bar{b}^{*M}_{i-1} \\ \sqrt{\lambda}\bar{\alpha}^{*d}_{M,i-2} & \bar{e}^{*M}_{d,i-1} \\ 0 & \bar{\gamma}^{1/2}_{i-1,M} \end{bmatrix} \bar{\Theta}^b_{M,i-1} = \begin{bmatrix} \bar{\Phi}^{b/2}_{M,i-1} & 0 \\ \bar{\alpha}^{*d}_{M,i-1} & \bar{e}^{*(M+1)}_{d,i-1} \\ b^M_{i-1} \bar{\Phi}^{-*b/2}_{M,i-1} & \bar{\gamma}^{1/2}_{i-1,M+1} \end{bmatrix}$$

where $e^M_{d,i-1} = \bar{e}^{*M}_{d,i-1} \bar{\gamma}^{1/2}_{i-1,M}$ is the a-posteriori joint-process estimation error due to estimating $d(i-1)$ from the data values $\{u(i-M), \ldots, u(i-1)\}$, $\bar{\alpha}^d_{M,i} = \bar{\theta}^d_{M,i} / \bar{\Phi}^{b/2}_{M,i}$, and $\bar{\theta}^d_{M,i} = \sum_{j=0}^{i} \lambda^{i-j} \bar{b}^{*M}_j \bar{e}^M_{d,j}$.

9. CONCLUDING REMARKS

We presented a unified derivation of different classes of adaptive filtering algorithms (in square-root form). This was done by embedding the standard exponentially weighted least-squares problem into an appropriate state-space model. Then the RLS, fast RLS, QR-, and lattice algorithms readily follow by proper identification with well-known algorithms in the linear least-squares estimation theory literature, such as the Riccati-based Kalman filter, the Chandrasekhar filter, the information filter, and the corresponding square-root versions. We further remark that most of the results described here extend smoothly to the multichannel, multidimensional, and nonlinear adaptive problems as we show in [6]. Moreover, the generalized shift structure allowed by the extended Chandrasekhar recursions allow us to consider more general windowing schemes, and extensions as discussed in [4, 5, 7].

REFERENCES

[1] M. L. Honig and D. G. Messerschmitt, *Adaptive Filters - Structures, Algorithms and Applications*, Kluwer Academic Publishers, 1984.

[2] S. Haykin, *Adaptive Filter Theory*, NJ:Prentice Hall, second edition, 1991.

[3] J. G. Proakis, C. M. Rader, F. Ling, and C. L. Nikias, *Advanced Digital Signal Processing*, NY:Macmillan Publishing Co., 1992.

[4] A. H. Sayed and T. Kailath, "A state-space approach to adaptive filtering," *submitted for publication*, 1992.

[5] A. H. Sayed, *Displacement Structure in Signal Processing and Mathematics*, PhD thesis, Stanford University, Stanford, CA, August 1992.

[6] B. H. Khalaj, A. H. Sayed, and T. Kailath, "A unified derivation of square-root multichannel least-squares filtering algorithms," in *Proc. of this conference.*

[7] B. Hassibi, A. H. Sayed, and T. Kailath, "Fast RLS adaptive filtering with alternative windowing schemes," *in preparation.*

[8] M. Morf and T. Kailath, "Square root algorithms for least squares estimation," *IEEE Transactions on Automatic Control*, vol. 20, no. 4, pp. 487–497, August 1975.

[9] A.H. Sayed and T. Kailath, "Structured matrices and fast RLS adaptive filtering," In *Proc. 2nd IFAC Workshop on Algorithms and Architectures for Real-Time Control*, Pergamon Press, pp. 211–216, Seoul, Korea, September 1992.

[10] A. H. Sayed and T. Kailath, "Extended Chandrasekhar recursions," to appear in *IEEE Transactions on Automatic Control.*

[11] G. Carayannis, D. Manolakis, and N. Kalouptsidis, "A fast sequential algorithm for least squares filtering and prediction," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol.31, no. 6, pp. 1394–1402, December 1983.

[12] J. Cioffi and T. Kailath, "Fast recursive-least-squares transversal filters for adaptive filtering," *IEEE Transactions on Acoustics, Speech and Signal Processing*, vol. 32, pp. 304–337, April 1984.

[13] A. Houacine, "Regularized fast recursive least squares algorithms for adaptive filtering," *IEEE Transactions on Signal Processing*, vo. 39, no. 4, pp. 860–870, April 1991.

[14] D. T. M. Slock, *Fast Algorithms for Fixed-Order Recursive Least-Squares Parameter Estimation*, PhD thesis, Stanford University, Stanford, CA, 1989.