

STRUCTURED MATRICES AND FAST RLS ADAPTIVE FILTERING*

ALI H. SAYED and T. KAILATH

Information Systems Laboratory, Stanford University, Stanford, CA 94305

Abstract. We present a new approach to the discrete-time Chandrasekhar recursions and some generalizations thereof. We extend the recursions to a class of structured time-variant state-space models, and discuss connections with the (generalized) Schur algorithm. We also apply the extended recursions to the adaptive filtering problem and give a transparent derivation of fast recursive least-squares algorithms.

Key Words. Recursive least squares, Adaptive systems, State-space methods, Kalman filters, Chandrasekhar recursions, Schur algorithm, Time-variant models.

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 (Kailath, 1981; Haykin, 1991). A wide range of algorithms and schemes has been developed to exploit the data structure and to reduce the computational complexity. These basically fall into three main groups of adaptive filtering algorithms: **Recursive Least Squares (RLS)** and the corresponding fast versions, **Lattice Least-Squares (LLS)**, and **QR-based least-squares** (see *e.g.*, (Haykin, 1991) and the references therein for more details on the subject). The derivation of the different schemes may be put into a unified framework (Sayed and Kailath, 1992) by relating the filtering problem to convenient structured (covariance) matrices. We focus here on the RLS approach and show that the associated fast versions follow by applying the (generalized) Schur algorithm to a Toeplitz-like covariance matrix. Fast RLS algorithms were originally derived in (Carayannis, Manolakis and Kalouptsidis, 1983; Cioffi and Kailath, 1984). It was later shown (Slock, 1989; Houacine, 1991) that these fast versions also follow by considering the Chandrasekhar equations (Kailath, Vieira and Morf, 1978) associated with the linear least-squares estimation problem. We present a new approach to the discrete-time Chandrasekhar recursions and some generalizations thereof. The approach is a simplification and extension of the deriva-

tion in (Kailath, Sayed and Lev-Ari, 1992). We extend the Chandrasekhar recursions to a class of *structured time-variant* state-space models, and show that within this framework, fast versions of the recursive least-squares algorithm in the single channel, multi-channel, nonlinear or multidimensional contexts, also follow naturally.

First, a brief review of known results. Consider a $p \times 1$ process $\{\mathbf{y}_i\}$ with an n -dimensional state-space model

$$\begin{aligned} \mathbf{x}_{i+1} &= F_i \mathbf{x}_i + G_i \mathbf{u}_i \\ \mathbf{y}_i &= H_i \mathbf{x}_i + \mathbf{v}_i \quad \text{for } i \geq 0 \end{aligned} \quad (1)$$

where $\{F_i, G_i, H_i\}$ are known matrices with dimensions $n \times n$, $n \times m$ and $p \times n$ respectively. We assume that \mathbf{x}_0 , \mathbf{u}_i and \mathbf{v}_i are stochastic variables that satisfy

$$E \mathbf{x}_0 = \bar{\mathbf{x}}_0, \quad E(\mathbf{x}_0 - \bar{\mathbf{x}}_0)(\mathbf{x}_0 - \bar{\mathbf{x}}_0)^* = \Pi_0$$

$$E \mathbf{u}_i \mathbf{x}_0^* = E \mathbf{v}_i \mathbf{x}_0^* = \mathbf{0}, \quad E \mathbf{v}_i = E \mathbf{u}_i = \mathbf{0}$$

$$E \begin{bmatrix} \mathbf{u}_i \\ \mathbf{v}_i \end{bmatrix} \begin{bmatrix} \mathbf{u}_j^* & \mathbf{v}_j^* \end{bmatrix} = \begin{bmatrix} Q_i & C_i \\ C_i^* & R_i \end{bmatrix} \delta_{ij}$$

with R_i positive-definite. The symbol $*$ denotes complex conjugation and the letter E denotes expected value. Let $\hat{\mathbf{x}}_{i|i-1}$ be the linear least-squares estimate of \mathbf{x}_i given $\{\mathbf{y}_0, \dots, \mathbf{y}_{i-1}\}$. The Kalman filter (Kalman, 1960) computes this quantity via the recursion: $\hat{\mathbf{x}}_{0|-1} = \bar{\mathbf{x}}_0$,

$$\hat{\mathbf{x}}_{i+1|i} = F_i \hat{\mathbf{x}}_{i|i-1} + K_i R_{\epsilon_i}^{-1} \epsilon_i \quad (2)$$

where $\epsilon_i = \mathbf{y}_i - H_i \hat{\mathbf{x}}_{i|i-1}$, $R_{\epsilon_i} = E(\epsilon_i \epsilon_i^*)$ and $K_i =$

* 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 is 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.

$E(\mathbf{x}_{i+1}\epsilon_i^*)$. Kalman showed that K_i and $R_{e,i}$ can be computed by the expressions

$$K_i = F_i P_{i|i-1} H_i^* + G_i C_i, \quad R_{e,i} = H_i P_{i|i-1} H_i^* + R_i$$

where $P_{i|i-1}$ is the error covariance in the one-step prediction of \mathbf{x}_i : $P_{i|i-1} = E(\mathbf{x}_i - \hat{\mathbf{x}}_{i|i-1})(\mathbf{x}_i - \hat{\mathbf{x}}_{i|i-1})^*$, and satisfies the Riccati difference recursion: $P_{0|-1} = \Pi_0$,

$$P_{i+1|i} = F_i P_{i|i-1} F_i^* - K_i R_{e,i}^{-1} K_i^* + G_i Q_i G_i^*$$

We can check that the number of operations (*i.e.* multiplications and additions) needed in going from index i to index $(i+1)$ in the Riccati recursion 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-invariant (also called constant-parameter) models $\{F, G, H, Q, R, C\}$. Indeed, it has been shown (see *e.g.*, (Kailath, Vieira and Morf, 1978)) that in the constant-parameter case the complexity can be reduced to $O(n^2\alpha)$ per iteration, where

$$\begin{aligned} \alpha &= \text{rank}(F\Pi_0 F^* + GQG^* - \bar{K}_{p,0}\bar{K}_{p,0}^* - \Pi_0) \\ &= \text{rank}(P_{1|0} - P_{0|-1}) \end{aligned}$$

and $\bar{K}_{p,i} = K_i R_{e,i}^{-1/2}$. This is achieved by using the so-called Chandrasekhar recursions to compute $\{K_i, R_{e,i}\}$ for use in the formulas (2). There are many forms for the Chandrasekhar recursions, but we shall give here perhaps the simplest (so-called square-root) version.

SQUARE-ROOT CHANDRASEKHAR ALGORITHM

Let $\delta P_i = P_{i+1|i} - P_{i|i-1}$. It turns out that for constant-parameter models, the quantity δP_i often has low rank, much less than n , and this fact can be exploited to find a lower complexity algorithm. Observe that δP_i is a Hermitian matrix, so that it has only real eigenvalues. We can factor it (nonuniquely) as

$$\delta P_i = P_{i+1|i} - P_{i|i-1} = L_i S_i L_i^* \quad (3)$$

where S_i is an $\alpha \times \alpha$ signature matrix, *viz.*, a diagonal matrix with as many ± 1 on the diagonal as δP_i has positive and negative eigenvalues. In fact we can assume, as shown ahead, that S_i is the same for all i (that is, $S_i = S \forall i$). We now form the pre-array (the reason for choosing such pre-arrays can be justified geometrically as in (Morf and Kailath, 1975)). They also arise naturally in the context of the Schur algorithm (Kailath, Sayed and Lev-Ari, 1992), as shown later in this paper. In this section, we shall give a simple algebraic verification.)

$$A_i = \begin{bmatrix} R_{e,i}^{1/2} & HL_i \\ \bar{K}_{p,i} & FL_i \end{bmatrix}$$

Let Θ_i be any $J = (I \oplus S_i)$ -unitary matrix ($\Theta_i J \Theta_i^* = J$) that triangularizes A_i . That is,

$$A_i \Theta_i = \begin{bmatrix} X & \mathbf{0} \\ Y & Z \end{bmatrix}$$

Comparing the entries on both sides of the equality $A_i J A_i^* = A_i \Theta_i J \Theta_i^* A_i^*$, we get

$$\begin{aligned} XX^* &= R_{e,i} + HL_i S_i L_i^* H^* \\ &\stackrel{(3)}{=} R_{e,i} + H(P_{i+1|i} - P_{i|i-1})H^* \\ &= R_{e,i} + R_{e,i+1} - R_{e,i} = R_{e,i+1} \end{aligned}$$

So we can choose $X = R_{e,i+1}^{1/2}$. Moreover,

$$\begin{aligned} YX^* &= K_i + FL_i S_i L_i^* H^* \\ &= K_i + F(P_{i+1|i} - P_{i|i-1})H^* = K_{i+1} \end{aligned}$$

and hence we can identify $Y = \bar{K}_{p,i+1}$. Finally, $YY^* + ZS_i Z^* =$

$$\begin{aligned} &= K_i R_{e,i}^{-1} K_i^* + FL_i S_i L_i^* F^* \\ &= K_i R_{e,i}^{-1} K_i^* + F(P_{i+1|i} - P_{i|i-1})F^* \end{aligned}$$

Therefore, $ZS_i Z^* = P_{i+2|i+1} - P_{i+1|i}$. Comparing with (3) we see that we can choose $S_{i+1} = S_i$ and $Z = L_{i+1}$. So we are led to the following recursion

$$\begin{bmatrix} R_{e,i}^{1/2} & HL_i \\ \bar{K}_{p,i} & FL_i \end{bmatrix} \Theta_i = \begin{bmatrix} R_{e,i+1}^{1/2} & \mathbf{0} \\ \bar{K}_{p,i+1} & L_{i+1} \end{bmatrix} \quad (4)$$

where Θ_i is any $J = (I \oplus S)$ -unitary matrix that produces a block zero in the (1,2) entry on the right-hand side of (4). These are called the *square-root Chandrasekhar recursions*. We can verify that each such iteration takes only $O((n+p)(n+\alpha)) = O(n^2\alpha)$ computations when $n \geq p$, as is often the case. We remark that the relation among the (generalized) Levinson algorithm, the (generalized) Schur algorithm and the Chandrasekhar recursions has been pointed out already in (Kailath, Vieira and Morf, 1978; Kailath, Sayed and Lev-Ari, 1992).

STRUCTURED TIME-VARIANT MODELS

The derivation of the Chandrasekhar recursions (4) is based on the fact that δP_i has low rank for constant-parameter systems, as shown in expression (3). We now show how to extend (4) to a class of *time-variant* state-space models that exhibit certain structure in their time-variation. For this purpose, we consider again the state-space model given by (1), and we shall say that it is a *structured* time-variant model if there exist $n \times n$ matrices Ψ_i such that F_i, G_i , and H_i vary according to the following rules:

$$H_i = H_{i+1} \Psi_i, \quad F_{i+1} \Psi_i = \Psi_{i+1} F_i, \quad G_{i+1} = \Psi_{i+1} G_i \quad (5)$$

It is clear that constant-parameter systems satisfy (5) with $\Psi_i = I$. We shall show later that another special case of (5) arises in the recursive least-squares problem. We further assume that the covariance matrices R_i, C_i and Q_i are time-invariant, even though the restrictions on Q_i, R_i and G_i can be dropped (Sayed and Kailath, 1992).

It turns out that in order to extend the Chandrasekhar recursions to time-variant structured models, we need to consider the difference $\delta_{\Psi_i} P_i = P_{i+1|i} - \Psi_i P_{i|i-1} \Psi_i^*$, which we factor (nonuniquely) as follows

$$P_{i+1|i} - \Psi_i P_{i|i-1} \Psi_i^* = L_i S_i L_i^* \quad (6)$$

where S_i is an $\alpha \times \alpha$ signature matrix. In fact, it also turns out that S_i is the same for all i ($S_i = S$). The reason for considering (5) and the difference (6) will become clear as soon as we give a simple algebraic verification of the proposed recursions, as well as when we discuss the connections with the (generalized) Schur algorithm. We now form the pre-array

$$A_i = \begin{bmatrix} R_{\epsilon, i}^{1/2} & H_{i+1} L_i \\ \Psi_{i+1} \bar{K}_{p, i} & F_{i+1} L_i \end{bmatrix}$$

and choose an arbitrary $J = (I \oplus S)$ -unitary matrix, Θ_i , that triangularizes A_i . That is,

$$A_i \Theta_i = \begin{bmatrix} X & \mathbf{0} \\ Y & Z \end{bmatrix}$$

Comparing the entries on both sides of the equality $A_i J A_i^* = A_i \Theta_i J \Theta_i^* A_i^*$, we can check easily, as done in the previous section, that $X = R_{\epsilon, i+1}^{1/2}$, $Y = \bar{K}_{p, i+1}$, and $Z = L_{i+1}$. For example, we have

$$\begin{aligned} X X^* &= R_{\epsilon, i} + H_{i+1} L_i S_i L_i^* H_{i+1}^* \\ &= R_{\epsilon, i} + H_{i+1} (P_{i+1|i} - \Psi_i P_{i|i-1} \Psi_i^*) H_{i+1}^* \\ &= R_{\epsilon, i} + H_{i+1} P_{i+1|i} H_{i+1}^* - H_i P_{i|i-1} H_i^* \\ &= R_{\epsilon, i+1} \end{aligned}$$

which shows that we can choose $X = R_{\epsilon, i+1}^{1/2}$. A similar argument holds for Y and Z . Therefore, we are led to the following (square-root) *extended Chandrasekhar recursion*

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

where Θ_i is *any* $J = (I \oplus S)$ -unitary matrix that produces a block zero in the (1,2) entry on the right-hand side of the above expression.

CONNECTION TO THE SCHUR ALGORITHM

Let $\mathcal{R} = E(\mathbf{y}_i \mathbf{y}_j^*)_{i,j=0}^\infty$ denote the covariance matrix of the output process $\{\mathbf{y}_i\}$, and define \mathcal{Z} to be the lower triangular shift matrix with ones on the p^{th} sub-diagonal and zeros elsewhere. Clearly \mathcal{R} is a Hermitian positive-definite block-matrix with $p \times p$ block-entries. We now verify that, for a structured time-variant state-space model as in (5), the covariance ma-

trix \mathcal{R} exhibits (displacement) structure, in the sense that $\mathcal{R} - \mathcal{Z} \mathcal{R} \mathcal{Z}^*$ has low rank. We then show that the (generalized) Schur algorithm reduces to the extended Chandrasekhar recursions when the extra structure provided by the assumed state-space model is properly incorporated into the Schur recursions. Let $\bar{\mathbf{x}}_i = E \mathbf{x}_i$, and define $\Pi_i = E(\mathbf{x}_i - \bar{\mathbf{x}}_i)(\mathbf{x}_i - \bar{\mathbf{x}}_i)^*$ to be the state covariance matrix. From the state equation (1) it follows that Π_i obeys the recursion $\Pi_{i+1} = F_i \Pi_i F_i^* + G_i Q G_i^*$. Moreover, the following simple identities are easy to check

$$\begin{aligned} \Pi_{i+1} - \Psi_i \Pi_i \Psi_i^* &= F^{[i]} \Delta F^{*[i]} \\ E(\mathbf{y}_i \mathbf{y}_i^* - \mathbf{y}_{i-1} \mathbf{y}_{i-1}^*) &= H_i F^{[i-1]} \Delta F^{*[i-1]} H_i^* \\ E(\mathbf{y}_i \mathbf{y}_{i+1}^* - \mathbf{y}_{i-1} \mathbf{y}_i^*) &= H_i F^{[i-1]} \Delta F^{*[i]} H_{i+1}^* \end{aligned}$$

where $F^{[i]} = F_i F_{i-1} \dots F_1$, $F^{[0]} = I$ and $\Delta = \Pi_1 - \Psi_0 \Pi_0 \Psi_0^*$. These expressions lead to $\mathcal{R} - \mathcal{Z} \mathcal{R} \mathcal{Z}^* =$

$$\begin{bmatrix} R_{\epsilon, 0} & K_0^* H_1^* & K_0^* F^{*[1]} H_2^* & \dots \\ H_1 K_0 & H_1 \Delta H_1^* & H_1 \Delta F^{*[1]} H_2^* & \\ H_2 F^{[1]} K_0 & H_2 F^{[1]} \Delta H_1^* & H_2 F^{[1]} \Delta F^{*[1]} H_2^* & \\ H_3 F^{[2]} K_0 & H_3 F^{[2]} \Delta H_1^* & H_3 F^{[2]} \Delta F^{*[1]} H_2^* & \\ \vdots & & & \ddots \end{bmatrix}$$

There is significant redundancy in the elements of $\mathcal{R} - \mathcal{Z} \mathcal{R} \mathcal{Z}^*$. To explore this, we factor the leading $2p \times 2p$ principal submatrix into the (rank and inertia revealing) form

$$\begin{bmatrix} R_{\epsilon, 0} & K_0^* H_1^* \\ H_1 K_0 & H_1 \Delta H_1^* \end{bmatrix} =$$

$$\begin{bmatrix} R_{\epsilon, 0}^{1/2} & \mathbf{0} \\ A & B \end{bmatrix} \begin{bmatrix} I & \mathbf{0} \\ \mathbf{0} & S \end{bmatrix} \begin{bmatrix} R_{\epsilon, 0}^{1/2} & \mathbf{0} \\ A & B \end{bmatrix}^*$$

for some signature matrix S and matrix entries A and B to be determined. Comparing both sides of the above expression we conclude that $A = H_1 \bar{K}_{p, 0}$ and $B S B^* = H_1 (P_{1|0} - \Psi_0 P_{0|-1} \Psi_0^*) H_1^*$. This suggests that we introduce the (nonunique) factorization $P_{1|0} - \Psi_0 P_{0|-1} \Psi_0^* = L_0 S L_0^*$, where L_0 is $n \times \alpha$ and S is an $\alpha \times \alpha$ signature matrix (notice that this derivation motivates the introduction of (6)). Hence we can take $B = H_1 L_0$. Moreover, we can now check that $\mathcal{R} - \mathcal{Z} \mathcal{R} \mathcal{Z}^* = \mathcal{G} J \mathcal{G}^*$, where $J = I \oplus S$ and

$$\mathcal{G} = \begin{bmatrix} R_{\epsilon, 0}^{1/2} & \mathbf{0} \\ H_1 \bar{K}_{p, 0} & H_1 L_0 \\ H_2 F^{[1]} \bar{K}_{p, 0} & H_2 F^{[1]} L_0 \\ H_3 F^{[2]} \bar{K}_{p, 0} & H_3 F^{[2]} L_0 \\ \vdots & \vdots \end{bmatrix}$$

We say that \mathcal{R} is a Hermitian close-to-Toeplitz matrix with respect to (\mathcal{Z}, J) , and \mathcal{G} is called a *generator* matrix. Notice that the rows of \mathcal{G} are closely related, which is a consequence of the underlying state-space

model. The block-triangular factorization of close-to-Toeplitz matrices, such as \mathcal{R} , can be computed efficiently (and recursively) by using the Schur reduction procedure (Lev-Ari and Kailath, 1986), which reduces to the following generator recursion: $\mathcal{G}_0 = \mathcal{G}$,

$$\begin{bmatrix} \mathbf{0}_p \\ \mathcal{G}_{i+1} \end{bmatrix} = \mathcal{G}_i \Theta_i \begin{bmatrix} \mathbf{0}_p & \\ & I_\alpha \end{bmatrix} + \mathcal{Z} \mathcal{G}_i \Theta_i \begin{bmatrix} I_p & \\ & \mathbf{0}_\alpha \end{bmatrix}$$

where Θ_i is a J -unitary matrix chosen such that the top p rows of \mathcal{G}_i (denoted by g_i) are reduced to the form $g_i \Theta_i = [r_i \ \mathbf{0}]$, where r_i is a $p \times p$ matrix. Therefore, the generator \mathcal{G}_{i+1} is obtained by multiplying \mathcal{G}_i by Θ_i and then shifting down the first p columns of $\mathcal{G}_i \Theta_i$ by p steps. We now apply this algorithm to the generator of \mathcal{R} . The first step involves multiplying by Θ_0 , which is the identity matrix since the first block-row of \mathcal{G}_0 already has a $p \times \alpha$ block zero, and shifting down the first block-column:

$$\mathcal{G}_1 = \begin{bmatrix} R_{\epsilon,0}^{1/2} & H_1 L_0 \\ H_1 \tilde{K}_{p,0} & H_2 F^{[1]} L_0 \\ H_2 F^{[1]} \tilde{K}_{p,0} & H_3 F^{[2]} L_0 \\ \vdots & \vdots \end{bmatrix}$$

Let Θ_1 be a J -unitary matrix such that

$$\begin{bmatrix} R_{\epsilon,0}^{1/2} & H_1 L_0 \end{bmatrix} \Theta_1 = \begin{bmatrix} X & \mathbf{0} \end{bmatrix}$$

Then we may check easily (as done before) that

$$\begin{bmatrix} R_{\epsilon,0}^{1/2} & H_1 L_0 \\ H_1 \tilde{K}_{p,0} & H_2 F^{[1]} L_0 \end{bmatrix} \Theta_1 = \begin{bmatrix} R_{\epsilon,1}^{1/2} & \mathbf{0} \\ H_2 \tilde{K}_{p,1} & H_2 L_1 \end{bmatrix}$$

where $P_{2|1} - \Psi_1 P_{1|0} \Psi_1^* = L_1 S L_1^*$. Therefore $\mathcal{G}_1 \Theta_1$ is equal to

$$\begin{bmatrix} R_{\epsilon,1}^{1/2} & \mathbf{0} \\ H_2 \tilde{K}_{p,1} & H_2 L_1 \\ H_3 F^{[2,2]} \tilde{K}_{p,1} & H_3 F^{[2,2]} L_1 \\ H_4 F^{[3,2]} \tilde{K}_{p,1} & H_4 F^{[3,2]} L_1 \\ \vdots & \vdots \end{bmatrix}$$

where $F^{[i,j]} = F_i F_{i-1} \dots F_j$. Next we shift down the first p columns, form Θ_2 and so on. We see that because of the special state-space structure of the elements of the generator of \mathcal{R} , there is significant redundancy in the generator array: the first two nonzero rows tell enough to fill out all other rows. That is exactly the (Chandrasekhar algorithm) simplification provided by the assumption of an underlying state-space model. So the basic recursion is just the ex-

tended Chandrasekhar algorithm:

$$\begin{bmatrix} R_{\epsilon,i}^{1/2} & H_{i+1} L_i \\ \Psi_{i+1} \tilde{K}_{p,i} & F_{i+1} L_i \end{bmatrix} \Theta_i = \begin{bmatrix} R_{\epsilon,i+1}^{1/2} & \mathbf{0} \\ \tilde{K}_{p,i+1} & L_{i+1} \end{bmatrix} \quad (7)$$

RECURSIVE LEAST-SQUARES

We now consider an important special case of (5) that arises in the recursive least-squares problem in adaptive filtering. The basic problem 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 ,

$$\mathbf{u}_i = [u_1(i) \ u_2(i) \ \dots \ u_M(i)] \quad (8)$$

($d(i)$ and $u_j(i)$, $j = 1, \dots, M$, are assumed scalar for

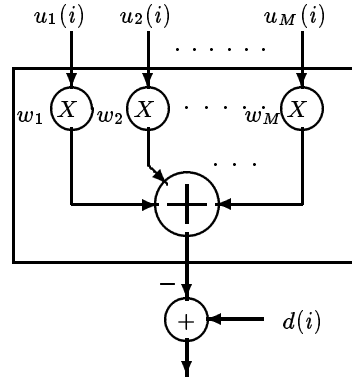


Fig. 1. Linear least-squares estimator.

simplicity), we are required to determine (recursively) the linear least-squares estimate of an $M \times 1$ column vector of unknown tap weights (see Fig. 1),

$$\mathbf{w} = [w_1 \ w_2 \ \dots \ w_M]^T,$$

so as to minimize the exponentially weighted error sum:

$$\min_{\mathbf{w}} \mathcal{E} = \min_{\mathbf{w}} \sum_{i=0}^N \lambda^{N-i} |d(i) - \mathbf{u}_i \mathbf{w}|^2, \quad 0 < \lambda \leq 1 \quad (9)$$

The parameter λ is often called the forgetting factor, since past inputs are (exponentially) weighted less than the more recent values. We may rewrite \mathbf{E} as follows

$$\mathcal{E} = \lambda^N \sum_{i=0}^N \left| \frac{d(i)}{(\sqrt{\lambda})^i} - \frac{\mathbf{u}_i \mathbf{w}}{(\sqrt{\lambda})^i} \right|^2,$$

which shows that minimizing \mathbf{E} is equivalent to the following minimization problem:

$$\min_{\mathbf{x}_0 = \mathbf{w}} \sum_{i=0}^N |y(i) - \mathbf{u}_i \mathbf{x}_i|^2 \quad (10)$$

where we defined the normalized quantities $y(i) =$

$d(i)/(\sqrt{\lambda})^i$ and $\mathbf{x}_i = \mathbf{w}/(\sqrt{\lambda})^i$. This simple manipulation of the expression for the total error shows that we can always rescale the original problem (9) (with an exponential factor λ) to another problem (10) with $\lambda = 1$. Now, the minimization in (10) may be easily recast into a Kalman filtering problem by considering the following M -dimensional state-space model

$$\begin{aligned} \mathbf{x}_{i+1} &= \lambda^{-1/2} \mathbf{x}_i, \quad \mathbf{x}_0 = \mathbf{w}, \quad \Pi_0 = \infty I \\ y(i) &= \mathbf{u}_i \mathbf{x}_i + v(i), \quad Ev(i)v^*(j) = \delta_{ij} \end{aligned} \quad (11)$$

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 $\hat{\mathbf{x}}_{i+1|i} = \mathbf{w}_i/(\sqrt{\lambda})^{i+1}$, which is given by $\hat{\mathbf{x}}_{0|-1} = \mathbf{0}$,

$$\hat{\mathbf{x}}_{i+1|i} = \lambda^{-1/2} \hat{\mathbf{x}}_{i|i-1} + \mathbf{k}_i r_{\epsilon,i}^{-1} \epsilon(i) \quad (12)$$

where $\epsilon(i) = y(i) - \mathbf{u}_i \hat{\mathbf{x}}_{i|i-1}$, $\mathbf{k}_i = \lambda^{-1/2} P_{i|i-1} \mathbf{u}_i^*$, $r_{\epsilon,i} = 1 + \mathbf{u}_i P_{i|i-1} \mathbf{u}_i^*$, and $P_{i|i-1}$ is the error covariance matrix that satisfies the Riccati difference equation

$$P_{i+1|i} = \lambda^{-1} [P_{i|i-1} - P_{i|i-1} \mathbf{u}_i^* r_{\epsilon,i}^{-1} \mathbf{u}_i P_{i|i-1}] \quad (13)$$

Recursions (12) and (13), along with the expressions for \mathbf{k}_i and $r_{\epsilon,i}$, constitute, apart from normalization, the well known *RLS algorithm*. Indeed, the Kalman filter variables $\{\mathbf{k}_i, r_{\epsilon,i}, P_{i+1|i}\}$ are essentially scaled versions of the RLS variables as usually described in the literature. To clarify this point, if we apply the matrix inversion lemma (see *e.g.*, (Kailath, 1980)) to (13), then we readily conclude that

$$\begin{aligned} P_{i+1|i}^{-1} &= \lambda P_{i|i-1}^{-1} + \lambda \mathbf{u}_i^* \mathbf{u}_i, \quad P_{0|-1}^{-1} = \mathbf{0} \\ &= \lambda \sum_{j=0}^i \lambda^{i-j} \mathbf{u}_j^* \mathbf{u}_j \equiv \lambda \Phi_i \end{aligned}$$

where Φ_i is usually referred to as the weighted auto-correlation matrix (Haykin, 1991). If we define the RLS variable $P_i = \Phi_i^{-1}$, then we get $P_i = \lambda P_{i+1|i}$. Moreover, the *a priori* error $e(i) = d(i) - \mathbf{u}_i \mathbf{w}_{i-1}$, is clearly related to the innovation by $\epsilon(i) = e(i)/(\sqrt{\lambda})^i$. The *a posteriori* error $e^p(i) = d(i) - \mathbf{u}_i \mathbf{w}_i$, may be written as

$$\begin{aligned} e^p(i) &= d(i) - (\sqrt{\lambda})^{i+1} \mathbf{u}_i \hat{\mathbf{x}}_{i+1|i} \\ &= e(i) [1 - \sqrt{\lambda} \mathbf{u}_i \mathbf{k}_i r_{\epsilon,i}^{-1}] = e(i) r_{\epsilon,i}^{-1} \end{aligned}$$

That is, the conversion factor γ_i , which converts the *a priori* error $e(i)$ into the *a posteriori* error $e^p(i)$, is given by $\gamma_i = r_{\epsilon,i}^{-1}$. We may finally, rewrite the Kalman filter equations in terms of the original RLS variables:

$$\begin{aligned} \mathbf{w}_i &= \mathbf{w}_{i-1} + \mathbf{g}_i e(i), \quad \mathbf{w}_{-1} = \mathbf{0} \\ \mathbf{g}_i &= \frac{\lambda^{-1} P_{i-1} \mathbf{u}_i^*}{1 + \lambda^{-1} \mathbf{u}_i P_{i-1} \mathbf{u}_i^*} \\ P_i &= \lambda^{-1} [P_{i-1} - \mathbf{g}_i \mathbf{u}_i P_{i-1}], \quad P_{-1} = \sigma I \end{aligned} \quad (14)$$

for $\sigma \gg 1$, and where we defined $\mathbf{g}_i = \sqrt{\lambda} \mathbf{k}_i r_{\epsilon,i}^{-1}$. In

summary, the Kalman variables and the RLS variables are related as in TABLE 1. We remark that we do not really need to rewrite the filter equations in terms of the original RLS variables, since computing $\hat{\mathbf{x}}_{i+1|i}$ is enough for determining \mathbf{w}_i . We established the correspondence between the Kalman and the RLS variables for the sake of comparison with the expressions that are usually described in the literature.

Table 1 Correspondence between the Kalman and RLS variables.

KF	RLS
$y(i)$	$d(i)/(\sqrt{\lambda})^i$
\mathbf{x}_i	$\mathbf{w}/(\sqrt{\lambda})^i$
$\hat{\mathbf{x}}_{i+1 i}$	$\mathbf{w}_i/(\sqrt{\lambda})^{i+1}$
$\lambda P_{i+1 i}$	$P_i = \Phi_i^{-1}$
$\sqrt{\lambda} \mathbf{k}_i r_{\epsilon,i}^{-1}$	\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 *non-zero* initial value \mathbf{w}_{-1} . This allows for instance, to change the tracking capability of the algorithm. It also allows to start the adaptive scheme with an initial guess obtained from a gradient-based method, such as the LMS algorithm (see (Haykin, 1991) for a related discussion). This change may be incorporated into our derivation by redefining the total error criterion (9):

$$\mathcal{E} = (\mathbf{w} - \bar{\mathbf{w}})^* \Pi_0^{-1} (\mathbf{w} - \bar{\mathbf{w}}) + \sum_{i=0}^N \lambda^{N-i} |d(i) - \mathbf{u}_i \mathbf{w}|^2$$

with $\bar{\mathbf{w}} = E\mathbf{w}$, $E(\mathbf{w} - \bar{\mathbf{w}})(\mathbf{w} - \bar{\mathbf{w}})^* = \Pi_0$. The state-space model (11) remains unchanged except for the new value of Π_0 . Moreover, we now use the filter equations (12–13) with the new initial conditions $P_{0|-1} = \Pi_0$ and $\hat{\mathbf{x}}_{0|-1} = \mathbf{w}_{-1} = \bar{\mathbf{w}}$ (we now have $P_{i+1|i}^{-1} = \lambda \Phi_i + \lambda^{i+1} \Pi_0^{-1}$).

FAST RECURSIVE LEAST SQUARES

We showed in the previous section that the RLS adaptive algorithm can be obtained by setting up a suitable state-space model (11), and by using the Riccati-based Kalman filter recursions (12–13). The state-space model we set up has special structure: $F = \lambda^{-1/2} I$, $G = 0$, $Q = 0$, and $R = 1$ are constant, while $H_i = \mathbf{u}_i$ is not. We now further assume that the input channels $\{u_1(\cdot), u_2(\cdot), \dots, u_M(\cdot)\}$ have a shift structure in the following sense: $u_j(i) = u_{j-1}(i-1)$. If we denote the value of the first channel at time i by $u(i)$, then this corresponds to having an input row \mathbf{u}_i of the form

$$\mathbf{u}_i = [u(i) \quad u(i-1) \quad \dots \quad u(i-M+1)] \quad (15)$$

The shift structure in \mathbf{u}_i suggests that we might be able to get fast RLS algorithms by using the extended Chandrasekhar recursions in place of the Riccati recursions. In fact this is true, and many results in the literature can be obtained in a more transparent (square-root array) form, and many variations and extensions derived in this way. To exploit the shift structure in \mathbf{u}_i , we consider the following $(N+1)$ - (not M -) dimensional state-space model

$$\begin{aligned} \mathbf{x}_{i+1} &= \lambda^{-1/2} \mathbf{x}_i, \quad \mathbf{x}_0 = \begin{bmatrix} \mathbf{w} \\ \mathbf{0} \end{bmatrix} \\ y(i) &= \mathbf{h}_i \mathbf{x}_i + v(i), \quad Ev(i)v^*(j) = \delta_{ij} \end{aligned} \quad (16)$$

where \mathbf{x}_i is now an $(N+1) \times 1$ state-vector with trailing zeros (added for convenience), and

$$\mathbf{h}_i = [u(i) \quad u(i-1) \quad \dots \quad u(0) \quad \mathbf{0}_{N-i}]$$

is a $1 \times (N+1)$ row vector. An initial state covariance matrix (with trailing zeros) is assumed, viz.,

$$E(\mathbf{x}_0 - \bar{\mathbf{x}}_0)(\mathbf{x}_0 - \bar{\mathbf{x}}_0)^* = \begin{bmatrix} \Pi_0 & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{bmatrix} \equiv \Pi_0 \oplus \mathbf{0}$$

where Π_0 is an $M \times M$ positive definite matrix. The Kalman equations of the previous section can now be rewritten as

$$\hat{\mathbf{x}}_{i+1|i} = \lambda^{-1/2} \hat{\mathbf{x}}_{i|i-1} + \mathbf{k}_i r_{\epsilon,i}^{-1} [y(i) - \mathbf{h}_i \hat{\mathbf{x}}_{i|i-1}]$$

$$r_{\epsilon,i} = 1 + \mathbf{h}_i P_{i|i-1} \mathbf{h}_i^*, \quad \mathbf{k}_i = \lambda^{-1/2} P_{i|i-1} \mathbf{h}_i^*$$

$$P_{i+1|i} = \lambda^{-1} [P_{i|i-1} - P_{i|i-1} \mathbf{h}_i^* r_{\epsilon,i}^{-1} \mathbf{h}_i P_{i|i-1}] \quad (17)$$

with $P_{0|-1} = \Pi_0 \oplus \mathbf{0}$. The gain vector $\bar{\mathbf{k}}_{p,i} = \mathbf{k}_i r_{\epsilon,i}^{-*/2}$ also has trailing zeros, $\bar{\mathbf{k}}_{p,i}^* \equiv [\mathbf{c}_i^* \quad \mathbf{0}]$, 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., $\mathbf{h}_i = \mathbf{h}_{i+1} Z$, may 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 $F_{i+1} Z = Z F_i$, since $F_i = \lambda^{-1/2} I$) shows that the state-space model (16) is indeed structured. The reduction in operation count can be now achieved by using the extended Chandrasekhar recursions (7) with $\Psi_i = Z$, $F_i = \lambda^{-1/2} I$, $G_i = 0$. To apply these recursions, we first introduce the (nonunique) factorization

$$L_0 S L_0^* = P_{1|0} - Z P_{0|-1} Z^* =$$

$$\lambda^{-1} \left(\begin{bmatrix} \Pi_0 & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{bmatrix} - \bar{\mathbf{k}}_{p,0} \bar{\mathbf{k}}_{p,0}^* \right) - Z \begin{bmatrix} \Pi_0 & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{bmatrix} Z^*$$

where L_0 and S are $(N+1) \times \alpha$ and $\alpha \times \alpha$ matrices respectively. The factor L_0 is of the form $L_0^* = [\tilde{L}_0^* \quad \mathbf{0}]$, where \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 . Writing down the extended Chandrasekhar recursions (7) we obtain

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

where Θ_i is any $J = (1 \oplus S)$ -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 fast RLS algorithms discussed in the literature (Carayannis, Manolakis and Kalouptsidis, 1983; Cioffi and Kailath, 1984). We note that the rotation Θ_i may be implemented in a variety of ways such as: elementary rotations, Householder transformations, square-root and/or division-free rotations, etc.. These may differ in computational complexity, numerical behaviour, and ease of hardware (VLSI) implementation. We also remark that explicit expressions for Θ_i can be written down in each case, leading to various explicit sets of equations.

In the prewindowed case $u(i) = 0$ for $i \leq 0$. This implies $\bar{\mathbf{k}}_{p,0} = \mathbf{0}$ and $P_{1/0} - Z P_{0/-1} Z^* =$

$$\lambda^{-1} \begin{bmatrix} \Pi_0 & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{bmatrix} - Z \begin{bmatrix} \Pi_0 & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{bmatrix} Z^* \quad (18)$$

A convenient choice is $\Pi_0 = \text{diag} \{ \lambda, \lambda^2, \dots, \lambda^M \}$, which leads to $\alpha = 2$, and we may choose $S = 1 \oplus -1$, and

$$\tilde{L}_0 = \begin{bmatrix} 1 & 0 & \dots & 0 & 0 \\ 0 & 0 & \dots & 0 & \lambda^{M/2} \end{bmatrix}^T$$

In the covariance case (where the past data $\{u(-M+1), \dots, u(-1), u(0)\}$ is assumed available), and for the same choice of Π_0 , we get $\alpha = 3$ (see (Sayed and Kailath, 1992) for more details). We also remark that other choices of Π_0 give other values of α .

EXTENSIONS AND CONCLUDING REMARKS

Recall that at the beginning of the last section we assumed that the channel inputs obey a shift structure, viz., $\mathbf{u}_j(i) = \mathbf{u}_{j-1}(i-1)$. This reflected into a special structured state-space model (16) with $\mathbf{h}_i = \mathbf{h}_{i+1} Z$. Our derivation, however, makes it clear that we can also obtain fast algorithms for other cases where the input channels do not necessarily exhibit a shift struc-

ture, the case usually considered in the literature. For example, if the input vector \mathbf{u}_i in (8) satisfies a relation of the form $\mathbf{u}_i = \mathbf{u}_{i+1}\Psi$, for some constant matrix Ψ , then the state-space model (11) is also structured, and we may write down the corresponding extended Chandrasekhar recursions. For matrices Ψ that are relatively sparse, in the sense that $\Psi\bar{k}_{p,i}$ requires $O(M)$ operations, and for appropriate choices of Π_0 , we are also led to a fast RLS algorithm. Moreover, the state-space model for the RLS problem ((11) or (16)) has a *row H* matrix, whereas our derivation allows for models with more general matrices H . Such models arise for example, in the multichannel, multidimensional and/or nonlinear adaptive problems, where in many instances, choice of Ψ with a block shift structure is convenient, such as: $\Psi = Z \oplus Z \oplus \dots \oplus Z$. The details of all these connections will be discussed elsewhere.

1. REFERENCES

- Kailath, T. (1981). *Lectures on Wiener and Kalman Filtering*, 2nd ed., Springer-Verlag, NY. 187 pp.
- Haykin, S. (1991). *Adaptive Filter Theory*, 2nd ed., Prentice Hall, Englewood Cliffs, NJ. 854 pp.
- Sayed, A. H. and T. Kailath (1992). A unified square-root derivation of adaptive filtering algorithms. Tentative title, to be submitted.
- Carayannis, G., D. Manolakis and N. Kalouptsidis (1983). A fast sequential algorithm for least squares filtering and prediction. *IEEE ASSP*, 31, 1394–1402.
- Cioffi, J. and T. Kailath (1984). Fast recursive-least-squares transversal filters for adaptive filtering. *IEEE ASSP*, 32, 304–337.
- Slock, D. T. M. (1989). *Fast Algorithms for Fixed-Order Recursive Least-Squares Parameter Estimation*. PhD thesis, Stanford University, CA.
- Houacine, A. (1991). Regularized fast recursive least squares algorithms for adaptive filtering. *IEEE SP*, 39, 860–870.
- Kailath, T., A. C. Vieira and M. Morf (1978). Orthogonal transformation (square-root) implementations of the generalized Chandrasekhar and generalized Levinson algorithms. *Lecture Notes in Control and Information Sciences*, 32, 81–91, Springer-Verlag, NY.
- Kailath, T., A. H. Sayed and H. Lev-Ari (1992). Generalized Chandrasekhar recursions from the generalized Schur algorithm. *IEEE AC*, submitted for publication (see also *Proc. ICASSP*, 4, 549–552, 1992).
- Kalman, R. E. (1960). A new approach to linear filtering and prediction problems. *Trans. ASME J. Basic Eng.* 82, 34–45.
- Morf, M. and T. Kailath (1975). Square root algorithms for least squares estimation. *IEEE AC* 20, 487–497.
- Lev-Ari, H. and T. Kailath (1986). Triangular factorization of structured Hermitian matrices. *Operator Theory: Advances and Applications* 18, 301–324, Birkhäuser, Boston.
- Kailath, T. (1980). *Linear Systems*. Prentice Hall Englewood Cliffs, NJ. 682 pp.