LATTICE FILTER INTERPRETATIONS OF THE CHANDRASEKHAR RECURSIONS FOR ESTIMATION AND SPECTRAL FACTORIZATION*

Ali H. Sayed [1], *Hanoch Lev-Ari* [2] *and Thomas Kailath* [1]

[1] Information Systems Laboratory, Stanford University, Stanford, CA 94305, USA.
[2] Dept. of Electrical and Computer Eng., Northeastern University, Boston, MA 02115, USA.

ABSTRACT

We use the classical Schur reduction procedure to give a lattice filter implementation of the Chandrasekhar recursions. The derivation is based on the observation that the covariance matrix of a process with time-invariant state-space model is structured. This allows us to derive easily the connection between the Schur algorithm and spectral factorization and to extend the Chandrasekhar recursions to the case of non-symmetric Riccati equations. We also remark that the Chandrasekhar recursions can be implemented in scalar steps using a sequence of well defined elementary (hyperbolic and Givens) rotations.

1. INTRODUCTION

We use the classical Schur reduction procedure [1] to give a lattice filter implementation of the Chandrasekhar recursions, which are known to provide a fast algorithm for the solution of the Riccati difference equation associated with least-squares estimation for constant-parameter state-space systems [2, 3]. The approach in this paper applies whether the observed signal is stationary or not and also gives a simple derivation of the connection between the Schur algorithm and spectral factorization. Recently there has been renewed interest in the subject, *e.g.* in [4] a connection between the Riccati equation and the matrix version of the Schur algorithm is pointed out. However the approach in [4] applies only to the stationary case and does not reveal clearly why such a connection exists. It should be noted that the relation between the (generalized) Levinson algorithm and the Chandrasekhar equations was shown much earlier [5].

* 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.

Our derivation reveals explicitly the connection between structured matrices, spectral factorization and Kalman filter theory. First, a brief review of known results. Consider a $p \times 1$ process $\{y_i\}$ with an n-dimensional time-invariant state-space model:

$$\begin{aligned} x_{i+1} &= Fx_i + Gu_i \\ y_i &= Hx_i + Dv_i \qquad \text{for } i \geq 0 \end{aligned}$$

where $\{F,\ G,\ H$ and $D\}$ are known matrices with dimensions $n \times n$, $n \times m$, $p \times n$ and $p \times q$ respectively. We assume that x_0, u_i, and v_i are stochastic variables with zero mean and satisfying

$$Ex_0x_0^* = \Pi_0$$

$$E\{\begin{bmatrix} u_i \\ v_i \end{bmatrix} \begin{bmatrix} u_j^* & v_j^* & x_0^* \end{bmatrix}\} = \begin{bmatrix} Q & C & \mathbf{0} \\ C^* & R & \mathbf{0} \end{bmatrix} \delta_{ij}$$

with R positive-definite. The symbol δ is the Kronecker delta function, $*$ denotes complex conjugation and the letter E denotes expected value. Let $\hat{x}_{i/i-1}$ and $\hat{y}_{i/i-1}$ denote the linear least squares estimates of x_i and y_i given $\{y_0,\ \ldots,\ y_{i-1}\}$ respectively. The innovation in y_i is defined by $e_i = y_i - \hat{y}_{i/i-1}$ and forms a white-noise innovations process [2] with $p \times p$ covariance matrix $R_i^e = Ee_ie_i^*$. We denote by $\{\nu_i\}$ the normalized innovation process obtained from $\{e_i\}$:

$$\nu_i = (R_i^e)^{-1/2}e_i$$

Let $\Pi_i = E(x_ix_i^*)$ denote the $n \times n$ state covariance matrix at time i. The Kalman filter [2] computes these quantities via the recursions:

$$\begin{cases} \hat{y}_{i/i-1} = H\hat{x}_{i/i-1} \\ \hat{x}_{i+1/i} = F\hat{x}_{i/i-1} + K_i(R_i^e)^{-1}e_i\ , \quad \hat{x}_{0/-1} = \mathbf{0} \\ \Pi_{i+1} = F\Pi_iF^* + GQG^* \end{cases}$$

where K_i is the $n \times p$ matrix defined by $K_i = Ex_{i+1}e_i^*$. It is easy to verify that K_i and R_i^e can be computed by the expressions

$$K_i = FP_iH^* + GCD^* \text{ and } R_i^e = HP_iH^* + DRD^*$$

where P_i is the error covariance in the one-step prediction of x_i:

$$P_i = E\left\{(x_i - \hat{x}_{i/i-1})(x_i - \hat{x}_{i/i-1})^*\right\}$$

and satisfies the Riccati difference recursion:

$$P_{i+1} = FP_iF^* - \tilde{K}_i\tilde{K}_i^* + GQG^* \quad (1)$$

$$P_0 = \Pi_0 \text{ and } \tilde{K}_i = K_i(R_i^e)^{-*/2}$$

In the stationary case (*i.e.* F stable and Π_0 is the solution of $\Pi_0 = F\Pi_0F^* + GQG^*$) R_i^e and K_i converge respectively to

$$R^e = HPH^* + DRD^* \text{ and } K = FPH^* + GCD^*$$

where P is the steady-state solution of the Riccati difference equation (1). It is well known [2] that the power spectrum $\mathcal{S}_y(z) = E\left[Y(z)Y^*(\frac{1}{z})\right]$ of the output process can be factored as $\mathcal{S}_y(z) = W(z)W^*(\frac{1}{z})$ where $W(z)$ is the transfer function from the normalized innovation process $\{\nu_i\}$ to the output process $\{y_i\}$

$$W(z) = (R^e)^{1/2} + H(zI-F)^{-1}\tilde{K}$$

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 $O(n^3)$ [2]. This complexity can be reduced to $O(n^2)$ in the case of time-invariant systems, where the Riccati difference equation can be replaced by the Chandrasekhar recursions [6]. We give a new derivation of this fact, showing that the recursions arise by combining state-space structures with the Schur algorithm for Toeplitz-related matrices (see, *e.g.* [1]).

2. DERIVATION OF THE CHANDRASEKHAR EQUATIONS

Let $\mathcal{R} = E(y_iy_j^*)_{i,j=0}^{\infty}$ denote the covariance matrix of the output process $\{y_i\}$ and define $\mathcal{Z}$ to be the lower triangular shift matrix with ones on the p^{th} subdiagonal. Clearly $\mathcal{R}$ is a Hermitian positive-definite block-matrix with $p \times p$ block-entries. The Chandrasekhar recursions exploit the fact that the process $\{y_i\}$ is the output of a time-invariant state-space model. In this case $\mathcal{R}$ turns out to be a structured matrix, in the sense that $\mathcal{R} - \mathcal{Z}\mathcal{R}\mathcal{Z}^*$ has low rank.

Fact (Some Useful Expressions) The following are valid identities:

$$\begin{aligned} \Pi_{i+1} - \Pi_i &= F^i\Delta F^{*i} \\ E(y_iy_i^* - y_{i-1}y_{i-1}^*) &= HF^{(i-1)}\Delta F^{*(i-1)}H^* \\ E(y_iy_{i+1}^* - y_{i-1}y_i^*) &= HF^{*(i-1)}\Delta F^{*i}H^* \end{aligned}$$

where $\Delta = \Pi_1 - \Pi_0$. ∎

These expressions lead to $\mathcal{R} - \mathcal{Z}\mathcal{R}\mathcal{Z}^* =$

$$\begin{bmatrix} R_0^e & K_0^*H^* & K_0^*F^*H^* & \cdots \\ HK_0 & H\Delta H^* & H\Delta F^*H^* & \\ HFK_0 & HF\Delta H^* & HF\Delta F^*H^* & \\ \vdots & & & \ddots \end{bmatrix}$$

There is significant redundancy in the elements of $\mathcal{R} - \mathcal{Z}\mathcal{R}\mathcal{Z}^*$ To explore this, we first factor the leading $2p \times 2p$ principal submatrix

$$\mathcal{R}_{2p} = \begin{bmatrix} R_0^e & K_0^*H^* \\ HK_0 & H\Delta H^* \end{bmatrix} \quad (2)$$

into the (rank and inertia revealing) form

$$\begin{bmatrix} (R_0^e)^{1/2} & 0 \\ A & B \end{bmatrix} \begin{bmatrix} I & 0 \\ 0 & -S \end{bmatrix} \begin{bmatrix} (R_0^e)^{1/2} & 0 \\ A & B \end{bmatrix}^* \quad (3)$$

for some signature matrix S and matrix entries A and B to be determined. Comparing (2) and (3) we conclude that $A = H\tilde{K}_0$ and $BSB^* = H(P_0 - P_1)H^*$. This suggests that we introduce the (nonunique) factorization $P_0 - P_1 = \tilde{L}_0S\tilde{L}_0^*$ where $\tilde{L}_0$ is an $n \times r$ matrix and S is the $r \times r$ signature matrix of $(P_0 - P_1)$, *i.e.* S has the form

$$S = \begin{bmatrix} -I_\gamma & 0 \\ 0 & I_\sigma \end{bmatrix}, \quad \gamma + \sigma = r$$

Hence we can take $B = H\tilde{L}_0$. Moreover, we can now check that we can write

$$\mathcal{R} - \mathcal{Z}\mathcal{R}\mathcal{Z}^* = \mathcal{G}J\mathcal{G}^*$$

where

$$J = \begin{bmatrix} I_p & 0 \\ 0 & -S \end{bmatrix} \text{ and } \mathcal{G} = \begin{bmatrix} (R_0^e)^{1/2} & 0 \\ H\tilde{K}_0 & H\tilde{L}_0 \\ HF\tilde{K}_0 & HF\tilde{L}_0 \\ \vdots & \vdots \end{bmatrix}$$

We say that $\mathcal{R}$ is a Hermitian near-Toeplitz block-matrix with respect to $(\mathcal{Z}, J)$ and $\mathcal{G}$ is called the generator matrix. The block-triangular factorization of such $\mathcal{R}$ can be computed efficiently (and recursively) by using the Schur reduction procedure [1, 7], which reduces to the following generator recursion:

$$\begin{bmatrix} 0_{p\times r} \\ \mathcal{G}_{i+1} \end{bmatrix} =$$

$$\mathcal{G}_i\Theta_i \begin{bmatrix} 0_p & \\ & I_r \end{bmatrix} + Z\mathcal{G}_i\Theta_i \begin{bmatrix} I_p & \\ & 0_r \end{bmatrix}, \quad \mathcal{G}_0 = \mathcal{G}$$

where Θ_i is a J-unitary matrix (*i.e.* $\Theta_i J \Theta_i^* = J$) chosen such that the top p rows of $\mathcal{G}_i$ (denoted by g_i) are reduced to the form $g_i\Theta_i = \begin{bmatrix} r_i & \mathbf{0} \end{bmatrix}$, 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 $p\times r$ block zero, and shifting down the first block-column:

$$\mathcal{G}_1 = \begin{bmatrix} (R_0^e)^{1/2} & H\tilde{L}_0 \\ H\tilde{K}_0 & HF\tilde{L}_0 \\ HF\tilde{K}_0 & HF^2\tilde{L}_0 \\ \vdots & \vdots \end{bmatrix}$$

We now define the $p \times r$, $p \times p$ and $r \times r$ matrices k_1, d_1 and q_1 respectively, such that:

$$\begin{cases} k_1 = (R_0^e)^{-1/2} H\tilde{L}_0 S \\ d_1 = (I_p - k_1 S k_1^*)^{-1/2} \\ q_1 S q_1^* = (S - k_1^* k_1)^{-1} \end{cases}$$

and consider the matrix Θ_1

$$\Theta_1 = \begin{bmatrix} I_p & -k_1 \\ -k_1^* & S \end{bmatrix} \begin{bmatrix} d_1 & \mathbf{0} \\ \mathbf{0} & q_1 \end{bmatrix}$$

We can verify easily that Θ_1 is a $J-$unitary matrix such that

$$\begin{bmatrix} (R_0^e)^{1/2} & H\tilde{L}_0 \\ H\tilde{K}_0 & HF\tilde{L}_0 \end{bmatrix} \Theta_1 = \begin{bmatrix} (R_1^e)^{1/2} & \mathbf{0} \\ H\tilde{K}_1 & H\tilde{L}_1 \end{bmatrix}$$

where $P_1 - P_2 = \tilde{L}_1 S \tilde{L}_1^*$. Therefore $\mathcal{G}_1\Theta_1$ is equal to

$$\begin{bmatrix} (R_1^e)^{1/2} & \mathbf{0} \\ H\tilde{K}_1 & H\tilde{L}_1 \\ HF\tilde{K}_1 & HF\tilde{L}_1 \\ \vdots & \vdots \end{bmatrix}$$

Next we shift down the first p columns, form Θ_2 and so on. In general, $k_i = (R_{i-1}^e)^{-1/2} H\tilde{L}_{i-1} S$, $d_i = (I_p - k_i S k_i^*)^{-1/2}$, $q_i S q_i^* = (S - k_i^* k_i)^{-1}$ and

$$\Theta_i = \begin{bmatrix} I_p & -k_i \\ -k_i^* & S \end{bmatrix} \begin{bmatrix} d_i & \mathbf{0} \\ \mathbf{0} & q_i \end{bmatrix}$$

This recursive procedure has the lattice filter interpretation shown in figure 1. 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. Therefore, we are led to the following square-root version of the Chandrasekhar equations [8]:

$$\begin{bmatrix} (R_{i+1}^e)^{1/2} & \mathbf{0} \\ \tilde{K}_{i+1} & \tilde{L}_{i+1} \end{bmatrix} = \begin{bmatrix} (R_i^e)^{1/2} & H\tilde{L}_i \\ \tilde{K}_i & F\tilde{L}_i \end{bmatrix} \Theta_i \quad (4)$$

We may note that the square-root form was first derived in [8], though by a very different method. We also remark that Θ_i can be implemented as a sequence of elementary (hyperbolic and Givens) rotations and hence it is possible to carry out the Chandrasekhar recursions in scalar steps [9]. Moreover, the derivation presented here can be easily extended to nonsymmetric Riccati equations of the form [9]:

$$P_{i+1} = FP_i\Phi^* + GQ\Gamma^* - K_i R_i^{-1} W_i^*$$

3. THE SCHUR ALGORITHM AND SPECTRAL FACTORIZATION

The previous discussion also leads to a simple derivation of the connection between spectral factorization and the Schur algorithm. Just observe that in the stationary case, equation (1) leads to $P_0 - P_1 = \tilde{K}_0\tilde{K}_0^*$. Hence we can take $\tilde{L}_0 = \tilde{K}_0$ and $S = I_p$, which shows that Θ_i reduces to the form

$$\begin{bmatrix} I_p & -k_i \\ -k_i^* & I_p \end{bmatrix} \begin{bmatrix} (I_p - k_i k_i^*)^{-\frac{1}{2}} & \mathbf{0} \\ \mathbf{0} & (I_p - k_i^* k_i)^{-\frac{1}{2}} \end{bmatrix}$$

where $\{k_i\}$ are now $p\times p$ contractive matrices, known as reflection coefficients. In this case, the recursive algorithm reduces to the celebrated Schur algorithm for matrix functions analytic and bounded inside the unit disc [10]. Moreover, these recursions give us the steady-state values of $\tilde{K}_i$ and $(R_i^e)^{1/2}$, which determine the spectral factor $W(z)$ as noted at the end of section 1. In fact we can say more. If we expand $W(z)$ and write

$$W(z) = (R^e)^{1/2} + \sum_{i=1}^{\infty} z^{-i} HF^{i-1}\tilde{K}$$

then the matrix coefficients of this expansion, viz., $\{(R^e)^{1/2},\ H\tilde{K},\ HF\tilde{K},\ HF^2\tilde{K},\ \ldots\}$, are exactly the

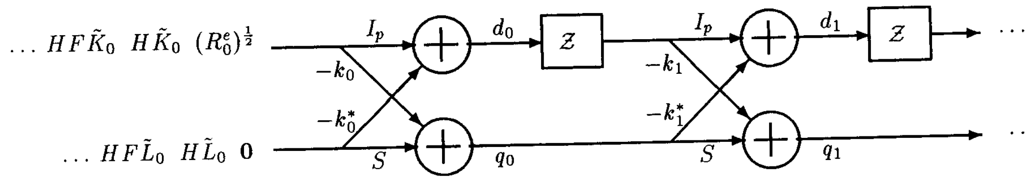

Figure 1: Lattice filter structure.

steady-state values of the first block-column of $\mathcal{G}_i\Theta_i$ (as $i \to \infty$). This relation is expected since the observations $\{y_0, \; y_1, \; y_2 \; \ldots\}$ and the normalized innovations $\{\nu_0, \; \nu_1, \; \nu_2, \; \ldots\}$ are related through the Cholesky factor of $\mathcal{R} = \mathcal{L}\mathcal{L}^*$ [2]:

$$[\; y_0^* \quad y_1^* \quad \ldots \;]^* = \mathcal{L} [\; \nu_0^* \quad \nu_1^* \quad \ldots \;]^*$$

and the block-columns of $\mathcal{L}$ are given by the first block-columns of $\mathcal{G}_i\Theta_i$ [1, 7]. Hence, in the steady state (as $n \to \infty$):

$$y_n = (R^e)^{1/2}\nu_n + \sum_{i=1}^{\infty} HF^{i-1}\tilde{K}\nu_{n-i}$$

which confirms the fact that the transfer function from the innovation process $\{\nu_i\}$ to the output process $\{y_i\}$ is $W(z)$.

4. CONCLUSION

We showed that the Chandrasekhar recursions follow by incorporating state-space structure into the Schur algorithm for structured matrices. We noted the simple connection to spectral factorization and also remarked that the approach of this paper can be extended to the nonsymmetric Riccati difference equation.

References

[1] H. Lev-Ari and T. Kailath, "Triangular factorization of structured Hermitian matrices," *Operator Theory: Advances and Applications*, I. Gohberg, Ed. Birkhäuser, Boston, vol. 18, pp. 301–324, 1986.

[2] T. Kailath, *Lectures on Wiener and Kalman Filtering*, NY: Springer-Verlag, second edition, 1981.

[3] R. E. Kalman, "A new approach to linear filtering and prediction problems," *Trans. ASME J. Basic Eng.*, vol. 82, pp. 34–45, 1960.

[4] C. Chang and T. T. Georgiou, "On a Schur based approach to spectral factorization – part II: Connection with the Riccati equation," *Linear Algebra and Its Applications*, to appear.

[5] B. Friedlander, T. Kailath, M. Morf, and L. Ljung, "Extended Levinson and Chandrasekhar equations for general discrete-time linear estimation problems," *IEEE Transactions on Automatic Control*, vol. AC–23, no. 4, pp. 653–659, August 1978.

[6] M. Morf, G. S. Sidhu, and T. Kailath, "Some new algorithms for recursive estimation in constant, linear, discrete-time systems," *IEEE Transactions on Automatic Control*, vol. AC–19, no. 4, pp. 315–323, August 1974.

[7] H. Lev-Ari and T. Kailath, "Lattice filter parametrization and modeling of nonstationary processes," *IEEE Transactions on Information Theory*, vol. IT–30, no. 1, pp. 2–16, January 1984.

[8] T. Kailath, A. C. Vieira, and M. Morf, "Orthogonal transformation (square-root) implementations of the generalized Chandrasekhar and generalized Levinson algorithm," in *Proc. IRIA Conference on Syst. Optimization and Analysis*, A. Bensoussarn and J. Lions, Eds. NY: Springer-Verlag, pp. 81–91, 1978.

[9] A. H. Sayed, H. Lev-Ari, and T. Kailath, "The generalized Schur algorithm in linear least-squares estimation," preprint 1991.

[10] Ph. Delsarte, Y. Genin, and Y. Kamp, "The Nevanlinna-Pick problem for matrix-valued functions," *SIAM J. Appl. Math.*, vol. 36, pp. 47–61, 1979.