Abstract

Recursive algorithms for the solution of linear least-squares estimation problems have been based mainly on state-space models. It has been known, however, that such algorithms exist for stationary time-series, using input-output descriptions (e.g., covariance matrices). We introduce a way of classifying stochastic processes in terms of their "distance" from stationarity that leads to a derivation of an efficient Levinson-type algorithm for arbitrary (nonstationary) processes. By adding structure to the covariance matrix, these general results specialize to state-space type estimation algorithms. In particular, the Chandra sekhar equations are shown to be the natural descendants of the Levinson algorithm.

1. Introduction

The problem of linear least squares estimation has been studied extensively and various methods of solution have been developed. These may be classified into estimation algorithms derived from input-output data or from other "external" system descriptions and algorithms derived from state-space or "internal" models. In the last decade the field of linear least-squares estimation has been dominated by state-estimation, in particular by the recursive Kalman-Bucy filter algorithms and its various versions, which rely heavily on the availability of state-space models.

In many applications, however, a state-space model is not readily available, and it would be preferable to have algorithms that use directly the covariance information of the observed process. The solution of the estimation problem is closely related to the problem of inverting the covariance matrix. Therefore, the computational efficiency of estimation algorithms is strongly dependent on the amount of computation required for inverting an appropriate matrix. For illustration, we shall mention the important example of a stationary process and its Toeplitz-type covariance matrix. It has been shown [1-3] that by making use of the special structure of the Toeplitz or the covariance matrix, it can be inverted in $O(N^2)$ operations (multiplications and additions), compared to $O(N^3)$ operations required in general for the inversion of an arbitrary matrix. It has also been long known in certain fields (e.g., geophysical data processing [4] and speech compression studies [5]) that recursive solutions can be obtained for the prediction of stationary processes. In particular, the so-called Levinson algorithm computes the optimal-predictor in $O(N^2)$ operations.

It has seemed in the past that the prediction of nonstationary processes would require $O(N^3)$ operations unless we can impose a state-space structure on the signal and noise processes. However, it is not unreasonable to expect that between the highly structured Toeplitz matrix (or stationary process) and a completely arbitrary covariance matrix, there should exist matrices (or processes) with varying degrees of structure and that this structure could be somehow utilized in reducing the amount of computation involved in the estimation problem. That this is indeed possible has been first shown in [6] by introducing the concept of "shift (low) rank," see also [7-9], and subsequently in [10,11]. These results were motivated for discrete-time problems by the work of Levinson [14] and Golub [18], and for continuous-time by the Chandra sekhar-type equations and their further developments in [12,13].

In this paper we shall introduce $\alpha$, an index of "distance from stationarity" of an arbitrary nonstationary process. We shall show how recursive solutions requiring of order $O(N^2)$ operations can be obtained for such processes with or without assuming a state-space structure. In the stationary case our solution reduces to known algorithms given in [14,15].

Finally, we shall show that, if the not necessarily stationary processes are known to come from state-space models, then this additional structural information can be used to reduce our general solution algorithm to the previously known Chandra sekhar-type equations. This means that we have been able to properly embed the state-space assumption into a general input-output framework.

2. A General Linear Estimation Problem

We shall consider the problem of estimating a stochastic process $x(\cdot)$ from observations of a related process $y(\cdot)$. Let $y(\cdot)$ (p-dimensional) and $x(\cdot)$ (n-dimensional) have covariance matrices

\begin{align*}
\mathbf{R}_y & = \mathbf{E}\{y(t)y(t)^\prime\} \\
\mathbf{R}_x & = \mathbf{E}\{x(t)x(t)^\prime\}
\end{align*}
The best linear least squares estimate of \( x_N \) given \( y_1, \ldots, y_N \) has the form
\[
\hat{x}(N|N-1) = \sum_{i=0}^{N-1} h_{xy}(N,1) y_1.
\]
The optimal one-step ahead predictor \( h_{xy}(N, \cdot) \) can be determined by using the well known orthogonality condition on the prediction error
\[
x_N - \hat{x}(N|N-1) = y_k, \quad 0 \leq k \leq N-1
\]
which means
\[
0 = E \left[ (x_N - \hat{x}(N|N-1)) y_k \right] = r_{nx,N-k} - \sum_{i=0}^{N-1} h_{xy}(N,1) r_{1,k},
\]
or in matrix form
\[
h_{xy}^N R_{xy}^{N-1} = \begin{bmatrix} r_{nx,0} & \ldots & r_{nx,N-1} \end{bmatrix},
\]
where \( h_{xy}^N = [h_{xy}(N,0), \ldots, h_{xy}(N,N-1)] \), an \( n \times Np \) matrix.

Note that by setting \( x = y \), we get an equation defining the predictor \( h(\cdot, \cdot) \) of the observed process itself, i.e.,
\[
\hat{y}(N|N-1) = \sum_{i=0}^{N-1} h(N,1) y_1,
\]
therefore
\[
h_{xy}^N R_{xy}^{N-1} = \begin{bmatrix} r_{ny,0} & \ldots & r_{ny,N-1} \end{bmatrix}.
\]
This can be rewritten as
\[
\begin{bmatrix} -h_{xy}^N \\ I \end{bmatrix} R_{xy}^N = \begin{bmatrix} 0, \ldots, 0, E \end{bmatrix},
\]
where \( E \) is determined by the left-hand side of this equation. For estimating \( x(\cdot) \) at a time instant within the observation interval \( (0, N) \), we have to find the optimal filter (smoother) \( h_{xy}(\cdot, \cdot; N) \), where
\[
\hat{x}(k|N) = \sum_{i=0}^{N} h_{xy}(k,1;N) y_1.
\]
Using the orthogonality condition again, now on the form
\[
x_k - \hat{x}(k|N) = y_1, \quad 0 \leq k \leq N
\]
we get
\[
\sum_{i=0}^{N} h_{xy}(k,1;N) r_{1,i} = r_{xy}^k, \quad 0 \leq k \leq N
\]
or in matrix form
\[
h_{xy}^N R_{xy}^N = \begin{bmatrix} r_{xy}^1 & \ldots & r_{xy}^N \end{bmatrix},
\]
where \( h_{xy}^N = [h_{xy}(1,0;N)] \), \( 0 \leq i, j \leq N \).

The last two matrix equations illustrate the fact that solving the estimation problem is closely related to the problem of inverting the covariance matrix, since for both solutions of the prediction- and the smoothing-problem we get
\[
\begin{bmatrix} s_{1,1} & \ldots & s_{1,N} \\ \vdots & \ddots & \vdots \\ s_{N,1} & \ldots & s_{N,N} \end{bmatrix} = \begin{bmatrix} s_{0,0} & \ldots & s_{0,N-1} \\ \vdots & \ddots & \vdots \\ s_{N-1,0} & \ldots & s_{N-1,N-1} \end{bmatrix}
\]
where \( s_{ij} \) are \( p \times p \) matrices (i.e., \( S \) is a block matrix). Then define
\[
[S] \triangleq \begin{bmatrix} s_{1,1} & \ldots & s_{1,N} \\ \vdots & \ddots & \vdots \\ s_{N,1} & \ldots & s_{N,N} \end{bmatrix} - \begin{bmatrix} s_{0,0} & \ldots & s_{0,N-1} \\ \vdots & \ddots & \vdots \\ s_{N-1,0} & \ldots & s_{N-1,N-1} \end{bmatrix}
\]
and
\[
[S] = \begin{bmatrix} s_{0,0} & \ldots & s_{0,N} \\ \vdots & \ddots & \vdots \\ s_{N-1,0} & \ldots & s_{N-1,N-1} \end{bmatrix} - \begin{bmatrix} 0 & \ldots & 0 \\ 0 & \ddots & \vdots \\ 0 & \ldots & 0 \end{bmatrix}
\]
We can now define the (block) displacement rank \( \alpha \) of matrix \( S \) as
\[
\alpha \triangleq \text{rank } [S]/p,
\]
where
\[
floor x \rfloor \triangleq \text{smallest integer } m, \text{ such that } m \geq n.
\]
We can now proceed in solving a general estimation problem by stating our assumptions on the crosscovariance \( \delta_{xy} \),
\[
\delta_{xy} = P_{xy} E P
\]
where \( P_{xy} \) is a \( N \times N \) matrix.

Note that if the processes \( x(t) \) and \( y(t) \) are jointly stationary, then
\[
\delta_{xy} = 0, \quad \delta_{xy} = 0, \quad P_{xy} = 0, \quad P = 0.
\]

The motivation behind this assumption is that in many problems the signal \( x(t) \) and the observations \( y(t) \) are connected by a linear relation of the form
\[
y_1 = Hx_1 + v_1 = x_1 + v_1
\]
where \( v(t) \) is white noise with unit intensity, uncorrelated with \( x(t) \). In this case,
\[
r_{1,j} = E_{y_1} y_j' = E_{x_1} x_j' + 1 \cdot \delta_{1,j} = E_{x_1} y_j' + 1 \cdot \delta_{1,j}
\]
\[
\delta_{1,j} = \begin{cases} 0 & i \neq j \\ 1 & i = j \end{cases}
\]
and \( R_{xy} \) will indeed satisfy the assumption (8).

As a matter of fact,
\[
R = \text{diag} \{ H \} R_{xy} + I
\]
and by operating with \( \delta_{xy} \) on both sides we get
\[
P = \text{diag} \{ H \} P_{xy}.
\]

4. The Levinson-Type Algorithm for the Joint \([x,y]\) Process

Using the assumptions stated in the previous section on the covari ance information, we can now give a set of recursions for computing \( h_{xy} \).
\[
h_{xy}^{m+1} = \begin{bmatrix} h_{xy}^m & 0 \end{bmatrix} + E M^{-1} M', \quad h_{xy}^1 = r_{xy} 1_0 00 (11a)
\]
\[
A_{xy}^{m+1} = \begin{bmatrix} 0 & B' \\ A & 0 \end{bmatrix} \begin{bmatrix} B' \\ A \end{bmatrix} = A^{-1} C', \quad A^0 = I (11b)
\]
\[
B_{xy}^{m+1} = \begin{bmatrix} B' \\ A \end{bmatrix} \begin{bmatrix} 0 & 0 \\ A & 0 \end{bmatrix} N^{-1} C', \quad B^0 = [1,0,...,0] (11c)
\]
\[
N_{xy}^{m+1} = N^{-1} C', \quad N_0 = r_{xy} 00 (12a)
\]
\[
M_{xy}^{m+1} = M^{-1} C', \quad M_0 = [r_{xy} 00] (12b)
\]
Define $\tilde{m} = (n+1)p$ and $\tilde{m} = mp$ (the size of $p^m$). The dimensions of $h_{xy}$, $A^m$, $B^m$, $N_m$, $M_m$, $F_m$, $C_m$ are $n \times \tilde{m}$, $(\tilde{m}+p) \times p$, $(\tilde{m}+p) \times \tilde{m}$, $p \times \tilde{m}$, $\tilde{m} \times \tilde{m}$, and $p \times \tilde{m}$, respectively. The quantities $F_m$, $C_m$ have to be computed at each step by

$$
F_m = \begin{bmatrix}
0 & 0 & \cdots & 0 & 0 \\
\vdots & \ddots & \ddots & \vdots & \vdots \\
0 & \cdots & 0 & 0 & 0 \\
\end{bmatrix} + \begin{bmatrix}
0 & 0 & \cdots & 0 & 0 \\
\vdots & \ddots & \ddots & \vdots & \vdots \\
0 & \cdots & 0 & 0 & 0 \\
\end{bmatrix}
$$

(13)

$$
C_m = \begin{bmatrix}
0 & 0 & \cdots & 0 & 0 \\
\vdots & \ddots & \ddots & \vdots & \vdots \\
0 & \cdots & 0 & 0 & 0 \\
\end{bmatrix} + \begin{bmatrix}
0 & 0 & \cdots & 0 & 0 \\
\vdots & \ddots & \ddots & \vdots & \vdots \\
0 & \cdots & 0 & 0 & 0 \\
\end{bmatrix}
$$

(14)

where $p_m$, $p_{xy}$ are the $n$th block row of $p$, $p_{xy}$, respectively, and

$$
p_m = \begin{bmatrix}
p_1 \\
p_2 \\
\vdots \\
p_m
\end{bmatrix}
$$

By counting the number of operations required at the $n$th step of the recursion, we get (assuming $p << \tilde{m}$ and ignoring terms accordingly) $(2n + 3p)\tilde{m}^2$ multiplications. Finding $h_{xy}$ will therefore require $(1.5 + n/p)\tilde{m}^2$ multiplications. The proof of the recursions above is somewhat lengthy and is given in Appendix A. It is however, important to note that the auxiliary quantities $A^m$, $B^m$ also obey the following equations.

$$
R^A_m = \begin{bmatrix}
0 \\
\vdots \\
0 \\
N_m
\end{bmatrix}
$$

(15)

$$
R^B_m = \begin{bmatrix}
1 & 0 \\
0 & 0 \\
0 & p^{-1}
\end{bmatrix}
$$

(16)

The first equation implies that

$$
A^m = \begin{bmatrix}
0 \\
\vdots \\
0
\end{bmatrix}
$$

so that $A^m$ is just the optimal predictor defined in the previous section. Note also that in the stationary case

$$
R^B_m = \begin{bmatrix}
0 \\
\vdots \\
0
\end{bmatrix}
$$

in which case $h^m$ is the smoothing filter for estimating $y_0$, given $(Y_1, 1 \leq i \leq N)$, or in this case also the so-called "backward predictor" [18, 20].

Recursive solutions of this type were derived for the stationary case by Levinson [11] for computing $h(\cdot, \cdot)$ and by Wiggins and Robinson [15] for computing $h_{xy}(\cdot, \cdot)$. Indeed, when we take $\alpha = 0$, $P$ = 0 the equations (11b), (11c), (12) reduce to the Levinson algorithm and (11), (12) can be shown to be equivalent to the equations of Wiggins and Robinson. Thus, the stationary case is nicely imbedded in our framework.

5. State Space Structure and Chandrasekhar-Type Equations

The results described so far are quite general and do not require a state space structure. We shall now show how by imposing more structure on the covariance matrices, the Chandrasekhar-type equations can be derived from the Levinson-type equation presented in the previous section.

Let $y(\cdot)$ and $x(\cdot)$ be the output and the state vectors of a linear system driven by white noise, i.e.,

$$
x_{i+1} = F_i x_i + u_i
$$

$$
y_i = H_i x_i + v_i
$$

$$
Ex_{i+1} y_i' = Q_{i+1} y_i', \\
Ev_{i+1} y_i' = R_{i+1} y_i'
$$

$$
Ex_{i+1} y_i' = 0
$$

In this case,

$$
Ex_{i+1} y_i' = F_i (Ex_{i+1} y_i') + Ev_{i+1} y_i',
$$

where the last term is zero for $i \geq i$. Therefore,

$$
x_{i+1} y_i = F_i x_i y_i' + I_{i+1} y_i',
$$

(18)

Also, as already noted earlier,

$$
x_{i+1} y_i = H_i x_i y_i' + I_{i+1} y_i'.
$$

In the following discussion, we shall therefore assume that $R$, $R_{xy}$ obey assumptions (18), (19) which are somewhat weaker than the state-space assumption above.

The optimal filter $h_{xy}(\cdot, \cdot)$ was shown to obey equation (1). Therefore,

$$
\sum_{i=0}^{t} h_{xy}(t+1, i) r_{i,s} = r_{t+1,s}
$$

$$
\sum_{i=0}^{s-1} h_{xy}(t, i) r_{i,s} = r_{t,s}
$$
Subtracting these last equations and using (18), we get
\[ \sum_{i=0}^{t-1} \left[ h_{xy}(t+1,1) - h_{xy}(t,1) \right] r_{i,s} + h_{xy}(t+1,t) r_{t,s} = x_{t+1,s} - x_{t,s} = (F_t - I) x_{t,s} \]
and from the Levinson recursions (11a), (11c) for \( h_{xy} \) and \( B \) we have
\[ h_{xy}(t+1,t) = h_{xy}(t,t-1) + E^{t-1}_t b^t, \]
\[ B_t = -N^{-1}_t \mathbf{c}_t - B_t^+ = \text{the last block row of } B_t, \]
so that
\[ h_{xy}(t+1,t) = h_{xy}(t,t-1) - E^{-1}_t E'_t H^{-1}_t N^{-1}_t. \] 

This recursion can be rewritten in another form that is easier to compare with the usual Chandrasekhar equations,
\[ h_{xy}(t+1,t) = h_{xy}(t,t-1) - F_t N^{-1}_t E'_t H^{-1}_t. \] 

The necessary algebra to derive this from (23) is given in Appendix B.

In Appendix C we shall also prove that
\[ E_t = (F - h_{xy}(t,t-1) H) E_{t-1}. \] 

Finally, note that (12), (24), (25) provide a complete set of recursions for computing \( h_{xy}(t+1,t) \), which are of the Chandrasekhar-type. Indeed, a comparison of our results to those presented in [16] shows that equations (24), (25), (12a), (12b) are precisely equations (16), (13'), (15), (14) of [16] if the following change of notation is made.

We see therefore that the Chandrasekhar equations are naturally induced by the Levinson-type recursions when the covariance matrices have a special structure. Furthermore, the parameter \( C_0 \) which appears in the Chandrasekhar algorithm can now be shown to have a meaning in terms of \( \bar{Q} \) (or \( \bar{Q} \)), the displacement rank of \( R \) (see also [3,9]).

To see this, let us recall that for a time-invariant state-space model
\[ F_{i,j} = \begin{cases} H_{i-j} \pi_j H' & 1 > j \\ H_{i,j} & 1 = j \end{cases} \] 
we have
\[ \pi_{i+1} = F_{i} P_{i} + Q \]
and
\[ r_{i+1,j} = H_{i-j} (\pi_{i+1} - \pi_j) H' \]
so that,
\[ r_{i+1,j} = H_{i-j} (\pi_{i+1} - \pi_j) H' \]
Writing this in matrix form gives

\[ \mathbf{Q}^N = 0 \begin{pmatrix} \mathbf{x}_1 - \mathbf{x}_0 \end{pmatrix} \mathbf{0} \]

where

\[ \mathbf{Q} = \begin{bmatrix} \mathbf{H} \\ \mathbf{H}^T \\ \vdots \\ \mathbf{H}^{N-1} \end{bmatrix} \]

is an extended observability matrix.

Let us take for simplicity the scalar case (i.e., \( p = 1, \mathbf{r}_{i,j} \) scalar). Assuming that the system is observable, \( \mathbf{0} \) will be a full rank matrix, and therefore

\[ \alpha = \text{rank } \mathbf{Q}^N = \text{rank } (\mathbf{x}_1 - \mathbf{x}_0) \]

Now the Chandrasekhar equations involve a parameter \( \alpha_0 \) defined as

\[ \alpha_0 = \text{rank } (\mathbf{x}_1 - \mathbf{x}_0 - \mathbf{F}_0 \mathbf{H}'(\mathbf{H}_0 \mathbf{H}' + \mathbf{I})^{-1} \mathbf{H}_0 \mathbf{F}') \]

and since the second term in the brackets is of rank 1,

\[ \alpha - 1 \leq \alpha_0 \leq \alpha + 1 \]

6. Conclusions

We have shown how recursive solutions can be obtained for the optimal predictor with covariance (or input-output) data, whether or not the state-space models are available. The complexity of these algorithms depends on a measure \( \alpha \) of the "distance" from stationarity of the signal and observed processes.

A similar approach makes it possible to derive a recursive solution for the optimal (filter) smoother \( \mathbf{h}_{xy} \). The details will not be presented here (see [10] for a partial treatment), but it is important to note that \( \mathbf{h}_{xy} \) can be computed in \( O(\alpha^2) \) operations, instead of \( O(\alpha^3) \) required for a direct solution of equation (4).

In the special case where the processes are known to come from a constant-parameter state-space model, the distance from stationarity \( \alpha \) coincides with a parameter describing the computational reductions obtainable by using the previously known [16] Chandrasekhar equations. Moreover, these general recursions reduce naturally to the Chandrasekhar equations in this and actually also in some more general cases. Note, for example, that we made no assumption on \( \mathbf{Q} \) and our derivation holds when it is time varying. Note also that our approach leads to a derivation of the Riccati equations that does not mention the Riccati equations, which was at the heart of the original derivation [16]. Actually, the general time-varying Riccati equation can also be embedded in the framework presented here (see [17]). Finally, we should note that these discrete-time results have close continuous-time analogs presented in [13] for the general problem of solving some integral equations, and in [12] dealing specifically with the estimation problem. In fact, it was these results that provided the immediate motivation for the discrete-time analysis presented here.

Appendix A

Proof of the Levinson-Type Algorithm for the Joint \((x,y)\) Process

The defining equation for \( h_{xy}^m \) was given as

\[ h_{xy}^m \mathbf{m}^{-1} = \begin{bmatrix} x_{\mathbf{m},0} & \cdots & x_{\mathbf{m},m-1} \end{bmatrix} \]

Therefore,

\[ \begin{bmatrix} 0 & h_{xy}^m \end{bmatrix} \mathbf{m}^{-1} = \begin{bmatrix} 0 & 0 & \cdots & 0 & r_{m,0} \end{bmatrix} + \begin{bmatrix} 0 & 0 & \cdots & 0 & r_{m,0} \end{bmatrix} \]

\[ \begin{bmatrix} 0 & r_{m,0} \cdots & r_{m,m-1} \end{bmatrix} + \begin{bmatrix} \alpha_0 & 0 & \cdots & 0 \end{bmatrix} \]

\[ \begin{bmatrix} 0 & r_{m,0} \cdots & r_{m,m-1} \end{bmatrix} + \begin{bmatrix} \alpha_0 & 0 & \cdots & 0 \end{bmatrix} + \begin{bmatrix} 0 & 0 & \cdots & 0 \end{bmatrix} \]

where

\[ \alpha_0 = \text{rank } \mathbf{Q}^N = \text{rank } (\mathbf{x}_1 - \mathbf{x}_0) \]

This can be rewritten as

\[ \begin{bmatrix} 0 & h_{xy}^m \end{bmatrix} \mathbf{m}^{-1} = \begin{bmatrix} 0 & 0 \end{bmatrix} \]
where the last block row of $A^n$ is the identity matrix.

$$R^{-1}B^n = \begin{bmatrix} 1 & 0 \\ 0 & 0 \\ \vdots & \vdots \\ 0 & p^{n-1} \end{bmatrix}$$ \hspace{1cm} (A4)

Note that from (A2) and (A4) it follows that

$$h_{xy}^{n+1} = \begin{bmatrix} 0 & h_{xy}^n \\ 0 & 0 \end{bmatrix} + E_{xy}^{-1}R^{n}$$ \hspace{1cm} (A5)

and from the defining equation, $h_{xy}^1 = F_{1y}^00^00^0$. Using a similar approach, the recursions for $A^n$, $B^n$ are derived.

$$R^{n+1} = \begin{bmatrix} \begin{bmatrix} 0 & 0 \\ \vdots & \vdots \\ 0 & 0 \end{bmatrix} + \begin{bmatrix} 0 \\ 1 \\ \vdots \\ 0 \\ 0 \end{bmatrix} \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \\ 0 \end{bmatrix} \begin{bmatrix} F_m \\ \vdots \\ F_m \\ 0 \\ 0 \end{bmatrix}$$ \hspace{1cm} (A6)

where

$$F_m = \begin{bmatrix} 0 & A_m \\ 0 & \vdots \\ 0 & 0 \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \\ 0 \end{bmatrix} \begin{bmatrix} F_m \\ \vdots \\ F_m \\ 0 \\ 0 \end{bmatrix}$$

$$R^{n+1} = \begin{bmatrix} \begin{bmatrix} 0 & 0 \\ \vdots & \vdots \\ 0 & 0 \end{bmatrix} + \begin{bmatrix} 0 \\ 1 \\ \vdots \\ 0 \\ 0 \end{bmatrix} \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \\ 0 \end{bmatrix} \begin{bmatrix} F_m \\ \vdots \\ F_m \\ 0 \\ 0 \end{bmatrix}$$ \hspace{1cm} (A7)

Combining the defining equations of $A^n$, $B^n$ and (A6), (A7) gives

$$A^{n+1} = \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \\ 0 \end{bmatrix} \begin{bmatrix} F_m \\ \vdots \\ F_m \\ 0 \\ 0 \end{bmatrix}, \hspace{1cm} A_0 = I.$$ \hspace{1cm} (A8)

$$N_{m+1} = N_m - \begin{bmatrix} k_m - [0,F_m] N_m \\ m ! m \\ \vdots \\ m ! m \end{bmatrix} M_{m}^{-1}P_m$$

$$= N_m - C_m M_{m}^{-1}P_m, \hspace{1cm} N_0 = r_0, 0.$$ \hspace{1cm} (A9)

where

$$C_m \triangleq k_m - [0,F_m] N_m.$$  

Also

$$B^{n+1} = \begin{bmatrix} \begin{bmatrix} 0 & 0 \\ \vdots & \vdots \\ 0 & 0 \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \\ 0 \end{bmatrix} \begin{bmatrix} F_m \\ \vdots \\ F_m \\ 0 \\ 0 \end{bmatrix} \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \\ 0 \end{bmatrix} \begin{bmatrix} F_m \\ \vdots \\ F_m \\ 0 \\ 0 \end{bmatrix}$$ \hspace{1cm} (A10)

and

$$M_{m+1} = M_m - F_m N_m^{-1}C_m, \hspace{1cm} M_0 = \begin{bmatrix} r_0, 0 \\ 0, 0 \end{bmatrix}.$$ \hspace{1cm} (A11)

To verify (A8), (A9), we premultiply the equations by $R^{n+1}$ and check that the right-hand side of those equations satisfy the defining equations (A3), (A4).

Finally, it remains to show that $C_m = F_m$. The proof is lengthy and shall be omitted. It can be found in [10, 11].

Appendix B

An Alternative form of the Recursion for $h_{xy}(t+1,t)$

$$h_{xy}(t+1,t) N_t = \left( h_{xy}(t,t-1) - F_{t-1}^H E_t H_{t-1}^{-1} H_t \right) \cdot \left( N_{t-1} - \left( E_{t-1}^H F_{t-1} E_{t-1} H_{t-1} \right)^{-1} \right) = h_{xy}(t,t-1) N_{t-1} - E_{t-1}^H F_{t-1} H_{t-1}^H$$

$$= h_{xy}(t,t-1) N_{t-1}$$ \hspace{1cm} (A12)

where

$$N_t = \begin{bmatrix} 0 & 1 \\ 0 & \vdots \\ 0 & 0 \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \\ 0 \end{bmatrix} \begin{bmatrix} F_m \\ \vdots \\ F_m \\ 0 \\ 0 \end{bmatrix}$$

In Appendix C, we shall show that

$$F_t = \left( F - h_{xy}(t,t-1) H_t \right) F_{t-1}$$

therefore

$$h_{xy}(t+1,t) N_t = h_{xy}(t,t-1) N_{t-1} - F_{t-1}^H E_{t-1}^H H_{t-1}.$$ \hspace{1cm} (A13)

Appendix C

The Recursion of $F_t$

As a first step, we shall show that
Using (C1), (C4), and (C5), we can rewrite (C3) as

\[
E_t = \begin{bmatrix}
F - h_{xy}(t, t-1) & h_{xy}(t, t-1)
\end{bmatrix}
\begin{bmatrix}
r_{t-1, 0} & p_{t-2, t} \\
F_{t-1, 0} & F_{t-1, 0} & p_{t-2, t}
\end{bmatrix}
\]

\[
= \begin{bmatrix}
F - h_{xy}(t, t-1) & h_{xy}(t, t-1)
\end{bmatrix}
\begin{bmatrix}
r_{t-1, 0} & p_{t-1, t}
\end{bmatrix}
\begin{bmatrix}
r_{t-1, 0} & p_{t-1, t} \\
F_{t-1, 0}
\end{bmatrix}
\]

\[
= (F - h_{xy}(t, t-1)) E_{t-1}
\]

\[\]
REFERENCES


**Title:** Levinson-and Chandrasekhar-type equations for a general, discrete-time linear estimation problem

**Authors:** B. Friedlander, M. Morf, T. Kailath and L. Ljung

**Performing Organization:** Stanford University

**Contract or Grant Number:** AFOSR-TR-77-0284

**Type of Report & Period Covered:** Interim

**Security Classification:** UNCLASSIFIED

**Distribution Statement:** Approved for public release; distribution unlimited.

**Abstract:** Recursive algorithms for the solution of linear least-squares estimation problems have been based mainly on state-space models. It has been known, however, that such algorithms exist for stationary time-series, using input-output descriptions (e.g., covariance matrices). We introduce a way of classifying stochastic processes in terms of their "distance" from stationarity that leads to a derivation of an efficient Levinson-type algorithm for arbitrary (nonstationary) processes. By adding structure to the covariance matrix, these general results specialize to state-space type estimation.

**Keywords:** Levinson-and Chandrasekhar-type equations, discrete-time linear estimation, state-space models, stationary time-series, covariance matrices.
20 Abstract

cont

algorithms. In particular, the Chandrasekhar equations are shown to be the natural descendents of the Levinson algorithm.