It follows that C is T-correcting for $T = \{(3,0),(2,2),(1,5),(0,9)\}$ or for $T = \{(2,2),(1,6),(0,11)\}$. Let us now consider two other ideals in A. I_3 is generated by $f(X)e(Y^{-1})$, and it has dimension 3 over GF(2); and I_4 is generated by $f(X)f(Y)$, and it has dimension 1 over GF(2). The code $C = I_1 + I_2 + I_3 + I_4$ is a (49,10) binary code, and its CDP is given by

$$b(C) = (20,14,7,0,0,0,0,0).$$

It is a T-correcting code for $T = \{(1,3),(0,9)\}$.

References

[1] F. J. MacWilliams and N. J. A. Sloane, The Theory of Error-Correcting Codes. New York: North-Holland, 1977.

[2] E. R. Berlekamp, Algebraic Coding Theory. New York: McGraw-Hill, 1968.

[3] P. Delsarte, "Automorphisms of Abelian codes," *Philips Res. Rep.*, vol. 25, pp. 389–402, 1970.

[4] F. J. MacWilliams, "Binary codes which are ideals in the group algebra of an Abelian group," *Bell Syst. Tech. J.*, vol. 49, pp. 987–1011, 1970.

[5] P. Piret, "Good block codes derived from cyclic codes," *Electron. Lett.*, vol. 10, no. 18, pp. 391–392, 1974.

[6] T. Krol, "The '(4,2)-concept' fault tolerant computer," in *Dig. 12th Int. Symp. Fault Tolerant Computing*, Santa Monica, CA, June 1982.

[7] E. N. Gilbert, "Capacity of a burst noise channel," *Bell Syst. Tech. J.*, vol. 39, pp. 1253–1256, 1960.

Recursive Limited Memory Filtering and Scattering Theory

ALFRED M. BRUCKSTEIN, MEMBER, IEEE, AND T. KAILATH, FELLOW, IEEE

Abstract—**Redheffer scattering theory is reviewed in a generalized setting as a method to derive recursive solutions of linear two-point boundary value problems (TPBVP) over arbitrarily varying intervals. The results can be used to derive a complete solution for the problem of limited-memory (or sliding-window) estimation, when a usual state-space model for the signal is available. Recursive limited-memory filters are derived for both continuous and discrete time signals.**

I. Introduction

Limiting the memory span of an estimation algorithm turns out to be a useful practice for solving problems of filter divergence due to mismodeling, for predicting signals with quasi-periodic components, and for detecting sudden and unexpected changes in systems generating the monitored signal (see, for example, Jazwinski [1] and Maybeck [2]). The issue of deriving recursive filters with limited memory, under the usual state-space assumptions, was addressed by several authors, most notably Schweppe [3] and Jazwinski [1]. Their efforts, however, concentrated on a particular case of interest, namely, the case of discrete signals with state-space models having no driving noise. Later work by Buxbaum [4] and Bierman [5] provided some alternative, computationally more advantageous solutions to the same problem. The "no driving noise" case is in fact a *parameter* estimation problem, with the unknown vector to be estimated being the initial state of the dynamic system. No solutions were ever given for the general case or for continuous signals (see, e.g., the remarks to this effect in Gelb [6, pp. 280–285] and also Maybeck [2, pp. 33–39].

Manuscript received March 16, 1983; revised March 9, 1984. This work was supported in part by the U.S. Army Research Office under Contract DAAG29-79-C-0215 and by the National Science Foundation under Grant ECS78-10003. The material in this correspondence was presented in part at the 16th Asilomar Conference on Circuits, Systems, and Computers, Pacific Grove, CA, November 1982.

The authors are with the Information Systems Laboratory, Electrical Engineering Department, Stanford University, Stanford, CA 94305, USA.

The limited-memory estimate is defined as the best linear estimate of the process (signal) of interest given, at time t, noisy observations over $[t-T, t)$. The problems in deriving estimation algorithms that provide such "sliding-window" estimates arise from the fact that one has to perform updates both to incorporate new data and to completely remove the effect of an observation in the past. To achieve this, Schweppe argued that the general sliding-window linear estimation algorithm should have the form

$$\text{new estimate} = \{\cdots\}\text{old estimate} + \{\cdots\}\text{data}(t) - \{\cdots\}\text{data}(t-T). \quad (1)$$

While it is true that in the case of no driving noise the limited memory algorithm can be cast in the above form, we shall show that in the general case some additional quantities enter into the update formulas, and these must be propagated as well.

Jazwinski obtained yet another type of limited-memory filter, which provides the estimate of the state (and of the signal) from quantities propagated by two growing-memory filters running in parallel, one estimating the process based on all the available data (up to t), the other based only on the data up to $t-T$. His solution, too, depends crucially on the no driving noise assumption, and it is not clear what modifications are required to derive the estimation algorithm for the general case.

In this correspondence we provide a complete solution to the limited-memory filtering problem, for the case of a general linear state-space model with driving noise. This solution is obtained through the scattering description of the state-space estimation problem (see for example [6]–[10]) and applies equally well to the continuous and discrete time cases. To our knowledge, the resulting algorithms are new. In fact, Maybeck [2] claims that it is preferable to solve the general problem by rerunning the growing-memory algorithm (Kalman filter) over the data interval *for each point in time* rather than obtaining the extremely complicated observation removal update. We shall show that this is not the case, since in the framework of scattering theory a complete solution of the problem is easily derived.

II. Problem Statement and the Hamiltonian Solution

Suppose that a state-space model of a continuous or discrete time signal $z(\cdot)$ is assumed to be

$$\nabla x(t) = A_t x(t) + B_t w(t)$$
$$z(t) = C_t x(t), \quad (2)$$

(where ∇ stands for either differentiation or the one-step time advance operator) and we are given noisy observations

$$y(t) = z(t) + v(t). \quad (3)$$

The driving and observation noises will be assumed to be uncorrelated white processes with intensities Q_t and R_t; we may note, however, that all the results can easily be extended to the correlated case.

Given observations over an interval $\Delta = [\tau_i, \tau_f]$, it is a well known result that the smoothed state estimates $\hat{x}(t|\Delta)$ are provided by the solution of a linear Hamiltonian two-point boundary value problem (TPBVP) as follows (see, e.g., [11]).

For the continuous case,

$$\frac{d}{dt}\begin{bmatrix}\hat{x}(t|\Delta)\\ \lambda(t|\Delta)\end{bmatrix} = \begin{bmatrix}A_t & B_tQ_tB_t^*\\ C_t^*R_t^{-1}C_t & -A_t^*\end{bmatrix}\begin{bmatrix}\hat{x}(t|\Delta)\\ \lambda(t|\Delta)\end{bmatrix} + \begin{bmatrix}0\\ -C_t^*R_t^{-1}y(t)\end{bmatrix} \quad (4a)$$

with boundary conditions

$$\lambda(\tau_f|\Delta) = 0, \qquad \hat{x}(\tau_i|\Delta) = x_i + P_i\lambda(\tau_i|\Delta). \tag{4b}$$

For the discrete case,

$$\begin{bmatrix} \hat{x}(t+1|\Delta) \\ \lambda(t|\Delta) \end{bmatrix} = \begin{bmatrix} A_t & B_tQ_tB_t^* \\ -C_t^*R_t^{-1}C_t & A_t^* \end{bmatrix}\begin{bmatrix} \hat{x}(t|\Delta) \\ \lambda(t+1|\Delta) \end{bmatrix} + \begin{bmatrix} 0 \\ C_t^*R_t^{-1}y(t) \end{bmatrix} \tag{5a}$$

with boundary conditions

$$\lambda(\tau_f + 1|\Delta) = 0, \qquad \hat{x}(\tau_i|\Delta) = x_i + P_i\lambda(\tau_i|\Delta). \tag{5b}$$

In the above boundary conditions, x_i and P_i summarize the prior knowledge on the state at the moment τ_i, i.e., the mean and variance of the initial state estimate with no observations. In the sequel we shall assume that no prior information is available; this is equivalent to formally setting $P_i \to \infty I$ in the above equations. This implies that the boundary conditions are zero for the adjoint variable at both endpoints of the data interval Δ. The corresponding estimates are the so-called "Fisher estimates" (see, e.g., [3]).

Now the problem of limited-memory estimation is to recursively determine $\hat{x}(t|\Delta_t)$ when the data interval is defined as a sliding window in time, i.e., $\Delta_t = [t - T, t)$.

Assume that the linear Hamiltonian system is solved by simple forward propagation of the extended state from some arbitrary initial condition. It is immediate that the corresponding extended final state will be given by the following formula:

$$\langle \text{ext state} \rangle_{\tau_f} = M^T \langle \text{ext state} \rangle_{\tau_i} + \Sigma^T, \tag{6}$$

where M^T is a transition matrix and the vector Σ^T summarizes the effect of nonzero input (both being obviously independent of the assumed initial extended state). If we have the pair $[M^T, \Sigma^T]$, the values of the solution of the Hamiltonian TPBVP at the boundaries are easily obtained, by simply setting the corresponding "adjoint" half of the initial and final extended state to zero. Therefore, in principle, all we need is to obtain recursively in time the pairs $[M^T(\Delta_t), \Sigma^T(\Delta_t)]$ or, as we shall see, some related quantities that are obtained by a Mason exchange rule from them.

Suppose that we are given the pair $[M^T, \Sigma^T]$ for some interval Δ_t and

$$[M^T, \Sigma^T] = \left[\begin{array}{cc|c} a & \beta & \sigma_u^T \\ b & \alpha & \sigma_d^T \end{array}\right]. \tag{7}$$

Then it is a straightforward algebraic exercise to show that the Fisher estimates are given by

$$\begin{bmatrix} \hat{x}(t|\Delta_t) \\ \hat{x}(t - T|\Delta_t) \end{bmatrix} = \begin{bmatrix} I & -ab^{-1} \\ 0 & -b^{-1} \end{bmatrix}\Sigma^T. \tag{8}$$

In the sequel we shall also deal with an exchanged matrix-vector pair associated to $[M^T, \Sigma^T]$ defined through a generalized "Mason exchange rule" as follows

$$[M^S, \Sigma^S] = \left[\begin{array}{cc|c} a - \beta\alpha^{-1}b & \beta\alpha^{-1} & \sigma_u^T - \beta\alpha^{-1}\sigma_d^T \\ -\alpha^{-1}b & \alpha^{-1} & -\alpha^{-1}\sigma_d^T \end{array}\right]. \tag{9}$$

This alternative and obviously equivalent description of the Hamiltonian system solution over Δ_t is useful since it embeds the error statistics for the resulting estimates. The entries of this alternative description, which will be called the "scattering" representation (as opposed to the original "transfer" representation), are exactly the variables that would have been propagated by a combined Kalman filter and a fixed-point smoother over any given data interval, yielding, when $P_i \to \infty I$, the corresponding Fisher estimates. (For a more detailed discussion of these results the reader is referred to [8]–[10].)

Denoting the block entries of the scattering representation by

$$[M^S, \Sigma^S] = \left[\begin{array}{cc|c} t & \rho & \sigma_u^S \\ r & \tau & \sigma_d^S \end{array}\right] \text{ or } \left[\begin{array}{cc|c} \Phi & P & U \\ +\Theta & \Psi & D \end{array}\right], \tag{10}$$

we have that (see [8] or [9])

$$\text{var}[x(t) - \hat{x}(t|\Delta_t)] = P + \Phi\Theta^{-1}\Phi^* \tag{11}$$

$$\text{var}[x(t-T) - \hat{x}(t - T|\Delta_t)] = \Theta^{-1}, \tag{12}$$

and the estimates are given by

$$\begin{bmatrix} \hat{x}(t(+1)|\Delta_t) \\ \hat{x}(t - T|\Delta_t) \end{bmatrix} = \begin{bmatrix} I & \Phi\Theta^{-1} \\ 0 & \Theta^{-1} \end{bmatrix}\Sigma^S. \tag{13}$$

We thus realize that a complete solution of the limited memory estimation problem calls for recursions directly providing either the scattering or the transfer representation of the Hamiltonian solution over sliding data intervals. It is precisely this problem that we can solve using the generalized Redheffer scattering theory.

III. Scattering Theory and the Propagation Algorithms

Redheffer developed scattering theory as a tool for the analysis of wave propagation through layered media [10]. We shall choose a presentation that emphasizes the fact that this theory generally deals with the evolution parameters of affine two-port systems under successive cascadings of infinitesimal or unit two-ports that we shall call "generators."

In the sequel, an affine two-port system will define a relationship between four n-vectors L_u, L_d, R_u, and R_d so that either the pair of left variables (L_u, L_d) or the left-upper and the right-lower variable (L_u, R_d) are considered as input or independent variables, the other two being the output vectors. We thus have in general

$$\langle \text{output vectors} \rangle = M \langle \text{input vectors} \rangle + \Sigma. \tag{14}$$

The $2n \times 2n$ system matrix M and the $2n$ internal source vector Σ completely define the two-port. If the inputs are the left variables, we have a *transfer* representation; otherwise we have a *scattering* representation. Fig. 1 illustrates these definitions.

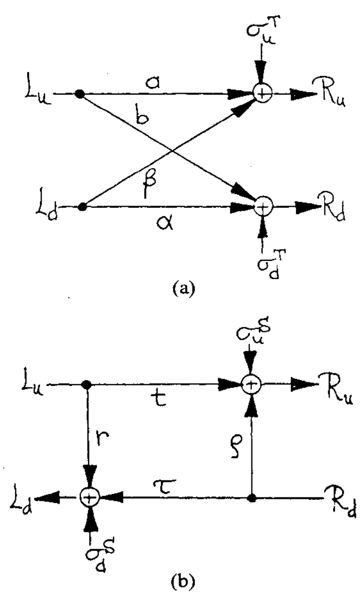

Fig. 1. Equivalent (a) transfer and (b) scattering representations of affine two-port.

Two different representations will be called equivalent if their defining matrix-vector pairs are related through the Mason exchange rule discussed in the previous section (9).

The cascade connection of two-ports in either representation spaces (denoted by ▶) is the system obtained by connecting (equating) the right variables of the first to the left variables of the second, upper to upper, and lower to lower. It is obviously an easy exercise in solving linear systems of equations to explicitly write down the composition rules that the cascading operation induces on the set of pairs $[M,\Sigma]$ or both representation domains. It turns out that we have *in the transfer domain*

$$[M_1^T,\Sigma_1^T]\blacktriangleright[M_2^T,\Sigma_2^T] = [M_2^TM_1^T, M_2^T\Sigma_1^T+\Sigma_2^T]; \quad (15)$$

in the scattering domain,

$$[M_1^S,\Sigma_1^S]\blacktriangleright[M_8^S,\Sigma_2^S] = [M_1^S * M_2^S, \Sigma_1^S \bullet \Sigma_2^S], \quad (16)$$

where $*$ is the Redheffer "star-product" and $\bullet$ stands for a rather complicated assembly sum. The explicit expressions defining these in terms of the blocks of the matrices and vectors involved are [10]

$$M_1^S * M_2^S = \begin{bmatrix} t_2(I-\rho_1 r_2)^{-1}t_1 & \rho_2 + t_2\rho_1(I-r_2\rho_1)^{-1}\tau_2 \\ r_1+\tau_1 r_2(I-\rho_1 r_2)^{-1}t_1 & \tau_1(I-r_2\rho_1)^{-1}\tau_2 \end{bmatrix} \quad (17a)$$

and

$$\Sigma_1^S \bullet \Sigma_2^S = \begin{bmatrix} \sigma_{u2}^S \\ \sigma_{d1}^S \end{bmatrix} + \begin{bmatrix} t_2(I-\rho_1 r_2)^{-1}(\sigma_{u1}^S + \rho_1\sigma_{d2}^S) \\ \tau_1(I-r_2\rho_1)^{-1}(\sigma_{d2}^S + r_2\sigma_{u1}^S) \end{bmatrix}. \quad (17b)$$

Although the composition rules in the two representation domains are very different, the underlying basic system-cascading structure induces a series of properties that are representation-independent. These properties are closure, associativity, the existence of a neutral (identity) system $[I,0]$, and (under certain conditions) the existence of an *annihilator* (or inverse) that, when cascaded to the given two-port, provides the identity element. Surprisingly, in spite of the radically different composition rules, the annihilator of $[M,\Sigma]$ turns out to be $[M^{-1}, -M^{-1}\Sigma]$ in *both representation domains* (provided, of course, that the system matrix is invertible). When the system matrix is singular, a transfer domain annihilator does not exist; however, if the system is in scattering representation, an annihilator may still exist if the transmission blocks (m_{11} and m_{22}) are nonsingular. This follows from a standard decomposition of any scattering representation as a cascade of three two-ports with lower-diagonal-upper system matrices (see [9]) and the fact that the inverse of a cascade is the cascade of inverses in reversed order.

An important issue that arises when dealing with continuous time results is the evolution under cascading with infinitesimal layers. Combining with a system of the form $[I+g\delta,\gamma\delta]$ when δ is infinitesimally small, yields the following results:

$$[M,\Sigma]\blacktriangleright[I+g\delta,\gamma\delta] = [M,\Sigma] + V\{M,\Sigma,g,\gamma\}\delta + o(\delta) \quad (18a)$$

$$[I+g\delta,\gamma\delta]\blacktriangleright[M,\Sigma] = [M,\Sigma] + \Lambda\{M,\Sigma,g,\gamma\}\delta + o(\delta), \quad (18b)$$

where V and Λ are representation-dependent functions. For the transfer domain these are simply

$$V_T\{M,\Sigma,g,\gamma\} = [gM, g\Sigma+\gamma] \quad (19a)$$

$$\Lambda_T\{M,\Sigma,g,\gamma\} = [Mg, M\gamma], \quad (19b)$$

and, as expected, in the scattering domain the composition laws are more complicated and involve the n-blocks of the arguments, as follows:

$$V_S\{[M^S,\Sigma^S],[g^S,\gamma^S]\} = \left[\begin{array}{cc|c} (m+\rho q)t & p+m\rho+\rho n+\rho q\rho & (m+\rho q)\sigma_u^S + \gamma_u + \rho\gamma_d \\ \tau q t & \tau(n+q\rho) & \tau(\gamma_d + q\sigma_u^S) \end{array}\right] \quad (20a)$$

$$\Lambda_S\{[M^S,\Sigma^S],[g^S,\gamma^S]\} = \left[\begin{array}{cc|c} t(m+pr) & tp\tau & t(\gamma_u + p\sigma_d^S) \\ q+rm+nr+rpr & (m+rp)\tau & (n+rp)\sigma_d^S + \gamma_d + r\gamma_u \end{array}\right], \quad (20b)$$

where

$$\left[\begin{array}{cc|c} m & p & \gamma_u \\ q & n & \gamma_d \end{array}\right]$$

is the block representation of $[g^S,\gamma^S]$. The above results are derived by simple $o(\delta)$ algebraic expansions of the basic cascade composition rules.

With the above preliminaries we can solve the following problem: Given a "generator sequence" $[I+g(\cdot)\delta, \gamma(\cdot)\delta]$, in continuous (infinitesimal δ) or discrete (unit δ) time, find recursions that provide $[M(\Delta_t),\Sigma(\Delta_t)]$, defined as the cascade of generators from $t-T$ up to t. This issue is important since, looking at the linear extended Hamiltonian system, we realize that the evolution of the extended state can be considered as the evolution of the equivalent internal source under successive cascadings with infinitesimal or unit layers that are completely characterized by the given state-space model and the observations. More explicitly, if the scattering domain generators are given by

$$[g(t),\gamma(t)] = \begin{cases} \left[\begin{array}{cc|c} A_t & B_tQ_tB_t^* & 0 \\ -C_t^*R_t^{-1}C_t & A_t^* & C_t^*R_t^{-1}y(t) \end{array}\right], & \text{continuous time} \\ \left[\begin{array}{cc|c} A_t - I & B_tQ_tB_t^* & 0 \\ -C_t^*R_t^{-1}C_t & A_t^*-I & C_t^*R_t^{-1}y(t) \end{array}\right], & \text{discrete time} \end{cases} \quad (21)$$

then the pair $[M(\Delta_t),\Sigma(\Delta_t)]$ provides all the quantities needed to compute the desired estimates, as we have seen in the previous section. If we had correlated driving and observation noise processes, the corresponding TPBVP's would yield slightly modified generator sequences; however, all the derivations would remain unchanged.)

The recursions for increasing t can be obtained as follows.

1) To go from $[M(\tau,t),\Sigma(\tau,t)]$ to $[M(\tau,t+\delta),\Sigma(\tau,t+\delta)]$ (where δ is infinitesimal in the continuous case and unity in the discrete case) requires a right cascading with the generator layer. Therefore we have

$$\nabla_t[M(\tau,t),\Sigma(\tau,t)] = \begin{cases} V\{M,\Sigma,g(t),\gamma(t)\}, & \text{continuous time} \\ [M,\Sigma]\blacktriangleright[I+g(t),\gamma(t)], & \text{discrete time} \end{cases} \quad (22)$$

These update equations, successively applied, are readily seen to be the usual growing-memory estimation algorithms (the Kalman filter and the associated fixed point smoother [8], [9]).

2) To obtain $[M(\tau+\delta,t),\Sigma(\tau+\delta,t)]$, a left cascade with the *annihilator* of the leftmost generator layer is needed. This gives

$$\nabla_\tau[M(\tau,t),\Sigma(\tau,t)] = \begin{cases} -\Lambda\{M,\Sigma,g(\tau),\gamma(\tau)\}, & \text{continuous time} \\ [I+g(\tau),\gamma(\tau)]^{-1}\blacktriangleright[M,\Sigma], & \text{discrete time} \end{cases} \tag{23}$$

In the above formulas, $\nabla.$ is the differentiation or advance operator corresponding to the argument in the subscript.

3) To complete one step of the limited-memory update, one has to obtain $\nabla_t\nabla_\tau[M,\Sigma]$ or $[M(\tau+\delta,t+\delta),\Sigma(\tau+\delta,t+\delta)]$. But it is readily seen that this simply requires combining the above steps, and this procedure is illustrated in Fig. 2.

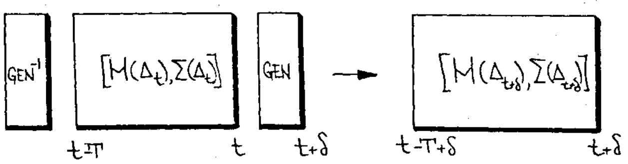

Fig. 2. Representation of sliding-memory update.

This provides complete recursions for $[M(\Delta_t),\Sigma(\Delta_t)]$. Of course, for initialization of sliding-window algorithms one first has to compute by a forward sweep the pair corresponding to Δ_0 and then set $\tau = t - T$.

Now, together with the results of the previous section, we have a complete solution of the limited-memory filtering problem. The estimates and their statistics are readily obtained from the recursive sliding-window system representations. Note that the recursions can be propagated in either the scattering or the transfer representation domains; however, to obtain the error statistics, one would prefer the scattering representation (and use (11)–(13)). It is also easily recognized that the above method is, in fact, a general technique for recursively solving linear TPBVP's over arbitrarily varying time intervals.

It is important to point out that in order to determine $[M(\Delta_t),\Sigma(\Delta_t)]$, one could also proceed as follows.

1) Propagate using forward (growing-memory) updates or $[M(0,t),\Sigma(0,t)]$ and store the result over an interval of depth T.

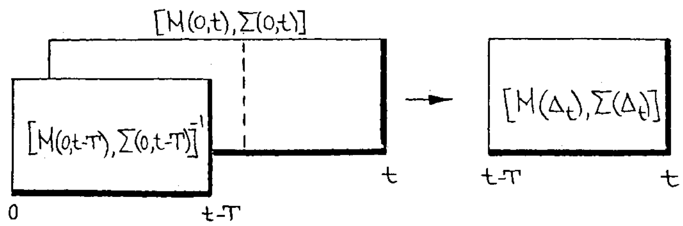

Fig. 3. Alternative method of sliding-window propagation.

2) compute the required two-port representation using the following relation (Fig. 3),

$$[M(\Delta_t),\Sigma(\Delta_t)] = [M(0,t-T),\Sigma(0,t-T)]^{-1}\blacktriangleright[M(0,t),\Sigma(0,t)]. \tag{24}$$

This solution is, in fact, more in the spirit of the algorithm proposed by Jazwinski, in which the estimates are computed from quantities propagated by two growing-memory filters [1]. For a comprehensive presentation of this material and a detailed listing of the algorithms, see [9].

IV. Conclusions

A complete solution for the general problem of limited-memory filtering for signals with given state-space models was provided in the framework of scattering theory. We note that in "given data" signal processing, sliding-window least-squares prediction algorithms are quite commonly applied (see, e.g., [12]). In such cases, however, due to the structure of the covariance estimators, a data-removing step is not more complicated than the usual time update formula. In the setting of general state-space models, scattering theory provides a similar result, i.e., that data removal is at least *in principle* identical to the step of incorporating new data points.

The above derived algorithms propagate a $2n\times 2n$ matrix and a $2n$ vector and provide through a further $n\times n$ matrix inversion the smoothed and filtered estimates together with their statistics. In the transfer domain the algorithm involves matrix multiplications, one $2n\times 2n$ matrix inversion and one $n\times n$ inversion per step. In the scattering domain, more complicated (Riccati-type) recursions are called for (see (20)). In some applications, when problems of modeling errors arise and relatively large computing power is available (as, for example, in satellite tracking [1]) these algorithms can be successfully implemented in order to avoid filter divergence.

When the signal model is time-invariant, the above solution provides a $2n$-dimensional time-invariant state-space filter/smoother. This result immediately follows from the fact that the representation matrix, depending on $g(\cdot)$ alone, remains constant (due to shift invariance in the cascading operation [9]). Thus, the limited-memory filter for the class of nonstationary processes with time-invariant state-space models (the initial conditions in the state-space model are arbitrary) is a constant parameter filter.

One issue of importance is the numerical stability of the resulting limited-memory filters. The numerical problems that arise (almost surely, since errors accumulate when the algorithm is propagated over long intervals) are easily solved by implementing a restart (reinitialization) procedure at intervals over which the results remain reliable. In this context an adaptive restart procedure can also be used, the idea being to increase or decrease the restart interval according to the value of an error measure computed at the previous reinitialization.

Also, to reduce the number of computations, one might implement an oscillating memory filter with memory span that grows from T to $2T$ and then is reset to T (by cascading with a corresponding medium annihilator). Such a method was first proposed by Jazwinski in connection with his two filter limited-memory algorithm.

References

[1] A. Jazwinski, *Stochastic Processes and Filtering Theory*. New York: Academic, 1970.
[2] P. Maybeck, *Stochastic Models, Estimation and Control*, vol. 2. New York: Academic, 1982.
[3] F. C. Schweppe, *Uncertain Dynamic Systems*. Englewood Cliffs, NJ: Prentice-Hall, 1973.
[4] P. J. Buxbaum, "Fixed-memory recursive filters," *IEEE Trans. Inform. Theory*, vol. IT-20, pp. 113–115, Jan. 1974.
[5] G. J. Bierman, "Fixed memory least squares filtering," *IEEE Trans. Inform. Theory*, vol. IT-21, pp. 690–692, Nov. 1975.
[6] A. Gelb, Ed., *Applied Optimal Estimation*. Cambridge, MA: Mass. Inst. Tech., 1974.
[7] T. Kailath, "Redheffer scattering and linear state-space estimation problems," *Richerche di Automatica*, vol. X/2, pp. 136–162, 1979.
[8] G. Verghese, B. Friedlander, T. Kailath, "Scattering theory and linear least squares estimation," *IEEE Trans. Automat. Cont.*, vol. AC-25, pp. 794–802, Aug. 1980.
[9] A. M. Bruckstein and T. Kailath, "The scattering formulation for state-space estimation and control problems," Inform. Syst. Lab. Rep., Stanford Univ., Stanford, CA, May 1982.
[10] R. Redheffer, "Difference equations and functional equations in transmission line theory," in *Modern Math for the Engineer*, E. F. Beckenbach, Ed. New York: McGraw-Hill, 1961, Ch. 12.
[11] A. E. Bryson and M. Frazier, "*Smoothing for linear and nonlinear dynamic systems*," *Proc. Conf. Optimum System Synthesis*, 1962; Tech. Rep. ASD-TDR-63-119, Aeronautical Syst. Div., Wright-Patterson AFB, Dayton, OH, 1963, pp. 354–364.
[12] G. C. Goodwin and R. L. Payne, *Dynamic System Identification: Experiment Design and Data Analysis. New York: Academic*, 1977.