Time-Variant Displacement Structure and Interpolation Problems

Ali H. Sayed, *Member, IEEE*, Tiberiu Constantinescu, and Thomas Kailath, *Fellow, IEEE*

***Abstract*— We derive a new recursive solution for a general time-variant interpolation problem of the Hermite–Fejér type, based on a fast algorithm for the recursive triangular factorization of time-variant structured matrices. The solution follows from studying the properties of an associated cascade system and leads to a triangular array implementation of the recursive algorithm. The system can be drawn as a cascade of first-order lattice sections, where each section is composed of a rotation matrix followed by a storage element and a tapped-delay filter. Such cascades always have certain blocking properties, which can be made equivalent to the interpolation conditions. We also illustrate the application of the algorithm to problems in adaptive filtering, model validation, robust control, and analytic interpolation theory.**

I. Introduction

The successful application of interpolation problems in control and circuit theory has inspired the study of generalizations to the time-variant setting. The extension of classical optimal control results to time-variant plants has been discussed by Feintuch and Francis [1], [2], and by Khargonekar *et al.* [3]–[5], and on the mathematical side there are precursors in the works Arveson [6] and Ball and Gohberg [7]. More complete analyses have been presented very recently. Alpay, Dewilde, and Dym [8], [9] developed a time-variant algebraic theory that is based on the so-called W-transform. This was then applied, along with generalizations of reproducing kernel Hilbert space techniques, to the solution of a time-variant (or nonstationary) version of the Nevanlinna–Pick problem. Ball *et al.* [11] addressed the same problem by using a formulation that is also based on the W-transform and on extensions of the theory described in [12].

In this paper we present a computationally-oriented solution for a general time-variant interpolation problem, based on the concept of time-variant displacement structure and on a fast generalized Schur-type algorithm for the recursive triangular factorization of a certain (implicitly defined) matrix. The point is that this recursive procedure can be associated with a time-variant "transmission-line" (a cascade of elementary time-variant sections), and transmission lines have certain (physically meaningful) "blocking" properties that make evident the family of all solutions to the interpolation problem. This readily allows us to treat a time-variant version of the Hermite–Fejér problem, which includes as a special case the aforementioned Nevanlinna–Pick problem. Results in this direction have been reported earlier in [29], [34]. We also remark that Dewilde and Dym have recently extended their W-transform-based approach beyond the Nevanlinna–Pick case to the solution of what they call a fundamental interpolation problem [10].

The present paper develops a new recursive approach, which has already been carried out in the time-invariant case in [13]–[16], where the picture is the following. Linear systems have transmission zeros: certain inputs at certain frequencies yield zero outputs. Each section of the cascade can be characterized by a $(p+q)\times(p+q)$ rational transfer matrix $T_i(z)$ say, that has a left zero-direction vector g_i at a frequency f_i, viz.,

$$g_i T_i(f_i) \equiv [u_i \quad v_i]\begin{bmatrix} T_{i,11} & T_{i,12} \\ T_{i,21} & T_{i,22} \end{bmatrix}(f_i) = \mathbf{0}$$

which makes evident (with the proper partitioning of the row vector g_i and the matrix function $T_i(z)$) the following interpolation property: $u_i T_{i,12}\, T_{i,22}^{-1}(f_i) = -v_i$. Hence, one way to think of solving an interpolation problem is to find a way of constructing a particular cascade from the given interpolation data. We shall extend this picture to the time-variant setting and describe the associated recursive solution. We shall also describe applications in adaptive filtering, robust control, model validation, and analytic interpolation theory.

The paper is organized as follows: after a brief introduction of our notation, we state in Section II the general time-variant Hermite–Fejér problem addressed in this paper. In Section III we derive necessary and sufficient conditions for the existence of solutions in terms of the positivity of a certain time-variant structured matrix $R(t)$. In Section IV we show that the structure of $R(t)$ can be exploited in deriving an efficient algorithm for its Cholesky factorization, and in Section V we show that the algorithm naturally leads to a cascade structure that parameterizes all solutions to the interpolation problem. In Section VI we further simplify the recursive algorithm and describe a triangular array implementation in terms of a cascade of lattice sections. In Section VII we discuss several applications whose solutions follow as special cases of the theory developed here. We first specialize the recursions to

Manuscript received September 17, 1992; revised June 21, 1993. Recommended by Past Associate Editor, E. W. Kamen. This work was supported in part by the Air Force Office of Scientific Research, Air Force Systems Command under Contract AFOSR91-0060 and by the Army Research Office under Contract DAAL03-89-K-0109 and by a fellowship from Fundação de Amparo à Pesquisa do Estado de São Paulo.

A. Sayed is with the Department of Electrical and Computer Engineering, University of California, Santa Barbara, CA 93106 USA.

T. Constantinescu is with the Programs in Mathematical Sciences, University of Texas at Dallas, Richardson, TX 75083 USA.

T. Kailath is with the Information Systems Laboratory, Stanford University, Standford, CA 94305 USA.

IEEE Log Number 9216627.

the time-invariant case leading to a computationally efficient procedure for the solution of analytic interpolation problems that arise in several areas in circuit theory and control. We then consider a model validation problem that arises in the context of validating uncertainty models in robust control and show how to solve it recursively by applying a special case of the general algorithm of this paper. A further extension of the Theorem 3.1 is also shown to solve the so-called strong Parrott's problem that arises in the study of the spectral properties of the four-block operator in control. As a final example, we consider the recursive least-squares problem that arises in adaptive filtering and control and show that the algorithm of this paper collapses to the well-known QR algorithm, but with the extra ingredient of providing a parallel method for the extraction of the weight vector.

A. Notation

We first introduce some notation. We consider a finite-dimensional linear time-variant state-space model with a bounded upper-triangular transfer operator $\mathcal{T}$. The matrix entries of $\mathcal{T}$ are denoted by T_{ij} (of dimensions $r(i) \times r(j)$) and correspond to the time-variant Markov parameters of the underlying state-space model

$$\mathcal{T} = \begin{bmatrix} \ddots & \ddots & & & & \\ & T_{-1,-1} & T_{-1,0} & T_{-1,1} & \cdots & \\ & & \boxed{T_{00}} & T_{01} & T_{02} & \cdots \\ & \mathbf{0} & & T_{11} & T_{12} & T_{13} \\ & & & & \ddots & \ddots \end{bmatrix} \tag{1}$$

where $\boxed{T_{00}}$ denotes the (0, 0) entry of $\mathcal{T}$. We further consider a stable sequence of scalar points $\{f(t)\}_{t\in\mathbf{Z}}$ ($\mathbb{Z}$ is the set of integers), viz., $\exists\, c > 0$ such that $|f(t)| < c < 1$ for all t. We also introduce the symmetric functions $s_k^{(n)}$ of n variables (taken k at a time). That is, $s_0^{(n)} = 1$ and

$$s_k^{(n)}(x_1, x_2, \cdots, x_n) = \sum_{1\le i_1<\cdots<i_k\le n} x_{i_1}x_{i_2}\cdots x_{i_k}.$$

For example, $n = 3$ corresponds to $s_0^{(3)} = 1$, $s_1^{(3)}(x_1, x_2, x_3) = x_1 + x_2 + x_3$, $s_2^{(3)}(x_1, x_2, x_3) = x_1x_2 + x_1x_3 + x_2x_3$, and $s_3^{(3)}(x_1, x_2, x_3) = x_1x_2x_3$.

For a uniformly bounded sequence of $1 \times r(t)$ row vectors $\{u(t)\}_{t\in\mathbf{Z}}$, viz., $\exists\, \bar{c} > 0$ such that $\|u(t)\| < \bar{c}$ for all t, we define the $1 \times r(t)$ row vector $u(t) \bullet \mathcal{T}(f(t))$ as

$$\begin{aligned} u(t) \bullet \mathcal{T}(f(t)) = u(t)T_{tt} &+ f(t)u(t-1)T_{t-1,\,t} \\ &+ f(t)f(t-1)u(t-2)T_{t-2,\,t} + \cdots . \end{aligned} \tag{2}$$

This corresponds to a time-variant tangential evaluation along the direction defined by $u(t)$. More generally, we define the $1 \times r(t)$ row vectors (for $p \ge 0$)

$$\begin{aligned} u(t) &\bullet \frac{1}{p!}\mathcal{T}^{(p)}(f(t)) \\ &\equiv \sum_{m=0}^{\infty} s_m^{(m+p)}[f(t),\, f(t-1), \cdots, f(t-m-p+1)] \\ &\cdot u(t-m-p)T_{t-m-p,\,t}. \end{aligned} \tag{3}$$

We shall also use the compact notation $[u_1(t)\ u_2(t)] \bullet \mathcal{H}_{\mathcal{T}}^2(f(t))$ to denote the row vector $[u_1(t) \bullet \mathcal{T}(f(t))\ \ u_1(t)\bullet \frac{1}{1!}\mathcal{T}^{(1)}(f(t)) + u_2(t) \bullet \mathcal{T}(f(t))]$, which we also write as

$$[u_1(t)\quad u_2(t)] \bullet \begin{bmatrix} \mathcal{T}(f(t)) & \frac{1}{1!}\mathcal{T}^{(1)}(f(t)) \\ \mathbf{0} & \mathcal{T}(f(t)) \end{bmatrix}.$$

More generally, we write $[u_1(t)\ u_2(t)\ \cdots\ u_r(t)] \bullet \mathcal{H}_{\mathcal{T}}^r(f(t)) =$ (see the equation at the bottom of the page). We finally denote by $e_i = [\mathbf{0}_{1\times i}\ 1\ \mathbf{0}]$ the ith basis vector of the n-dimensional space of complex numbers $\mathbf{C}^{1\times n}$.

II. The Time-Variant Hermite–Fejér Problem

We now state the general Hermite–Fejér interpolation problem treated in this paper. We consider m stable time-variant points $\{\alpha_i(t)\}_{i=0}^{m-1}$, and we associate with each point $\alpha_i(t)$ a positive integer $r_i \ge 1$ and uniformly bounded row vectors $\mathbf{a}_i(t)$ and $\mathbf{b}_i(t)$ partitioned as

$$\mathbf{a}_i(t) = \begin{bmatrix} u_1^{(i)}(t) & u_2^{(i)}(t) & \cdots & u_{r_i}^{(i)}(t) \end{bmatrix} \quad \text{and}$$

$$\mathbf{b}_i(t) = \begin{bmatrix} v_1^{(i)}(t) & v_2^{(i)}(t) & \cdots & v_{r_i}^{(i)}(t) \end{bmatrix}$$

where $u_j^{(i)}(t)$ and $v_j^{(i)}(t)$ ($j = 1, \cdots, r_i$) are $1 \times p(t)$ and $1 \times q(t)$ row vectors, respectively. That is, $\mathbf{a}_i(t)$ and $\mathbf{b}_i(t)$ are partitioned into r_i row vectors each. The time-variant tangential Hermite–Fejér interpolation problem then reads as follows.

Problem 2.1: Given m stable points $\{\alpha_i(t)\}$, with the associated uniformly bounded data $\mathbf{a}_i(t)$ and $\mathbf{b}_i(t)$, describe all upper-triangular strictly contractive transfer operators $\mathcal{S}$ ($\|\mathcal{S}\|_\infty < 1$) that satisfy

$$\mathbf{b}_i(t) = \mathbf{a}_i(t) \bullet \mathcal{H}_{\mathcal{S}}^{r_i}(\alpha_i(t)) \qquad \text{for } 0 \le i \le m-1 \text{ and } t \in \mathbb{Z}. \tag{4}$$

■

It is clear that the above Hermite–Fejér problem includes, among others, several important special cases such as:

- *Scalar time-variant Carathéodory–Fejér*: $m = 1$, $\alpha_0(t) = 0$, $r_0 = n$, $p(t) = q(t) = 1$, $\mathbf{a}_0(t) = [1\ 0\ \cdots\ 0]$,

$$[u_1(t)\quad u_2(t)\quad \cdots\quad u_r(t)] \bullet \begin{bmatrix} \mathcal{T}(f(t)) & \frac{1}{1!}\mathcal{T}^{(1)}(f(t)) & \frac{1}{2!}\mathcal{T}^{(2)}(f(t)) & \cdots & \frac{1}{(r-1)!}\mathcal{T}^{(r-1)}(f(t)) \\ & \mathcal{T}(f(t)) & \frac{1}{1!}\mathcal{T}^{(1)}(f(t)) & \cdots & \frac{1}{(r-2)!}\mathcal{T}^{(r-2)}(f(t)) \\ & & \ddots & & \vdots \\ & \mathbf{0} & & \mathcal{T}(f(t)) & \frac{1}{1!}\mathcal{T}^{(1)}(f(t)) \\ & & & & \mathcal{T}(f(t)) \end{bmatrix}.$$

$\mathbf{b}_0(t) = [\beta_0(t)\ \beta_1(t)\ \cdots\ \beta_{n-1}(t)]$. In this case, we are reduced to finding upper-triangular strict contractions $\mathcal{S}$ (with scalar entries s_{ij}) that satisfy $s_{tt} = \beta_0(t)$, $s_{t-1,t} = \beta_1(t), \cdots, s_{t-n-1,t} = \beta_{n-1}(t)$, or equivalently, $\frac{1}{i!}\mathcal{S}^{(i)}(0(t)) = \beta_i(t)$ for $i = 0, 1, \cdots, n-1$, where $0(t) = 0$ for all $t \in \mathbb{Z}$.

- *Scalar time-variant Nevanlinna–Pick:* $m = n$, $\alpha_i(t)$ stable, $r_i = 1$, $p(t) = q(t) = 1$, $\mathbf{a}_i(t) = 1$, $\mathbf{b}_i(t) = \beta_i(t)$. In this case, we are reduced to finding upper-triangular strict contractions $\mathcal{S}$ (with scalar entries s_{ij}) that satisfy $\mathcal{S}(\alpha_i(t)) = \beta_i(t)$ for $i = 0, 1, \cdots, n-1$.

III. Solvability Condition

The first step in the solution consists in constructing three matrices $F(t)$, $G(t)$, and $J(t)$ directly from the interpolation data: $F(t)$ contains the information relative to the points $\{\alpha_i(t)\}$ and the dimensions $\{r_i\}$, $G(t)$ contains the information relative to the direction vectors $\{\mathbf{a}_i(t), \mathbf{b}_i(t)\}$, and $J(t) = (I_{p(t)} \oplus -I_{q(t)})$ is a signature matrix, where $I_{p(t)}$ denotes the $p(t) \times p(t)$ identity matrix. The matrices $F(t)$ and $G(t)$ are constructed as follows: we associate with each $\alpha_i(t)$ a Jordan block $\overline{F}_i(t)$ of size $r_i \times r_i$

$$\overline{F}_i(t) = \begin{bmatrix} \alpha_i(t) & & & \\ 1 & \alpha_i(t) & & \\ & \ddots & \ddots & \\ & & 1 & \alpha_i(t) \end{bmatrix}$$

and two $r_i \times p(t)$ and $r_i \times q(t)$ matrices $U_i(t)$ and $V_i(t)$, respectively, which are composed of the row vectors associated with $\alpha_i(t)$

$$U_i(t) = \begin{bmatrix} u_1^{(i)}(t) \\ u_2^{(i)}(t) \\ \vdots \\ u_{r_i}^{(i)}(t) \end{bmatrix} \quad \text{and} \quad V_i(t) = \begin{bmatrix} v_1^{(i)}(t) \\ v_2^{(i)}(t) \\ \vdots \\ v_{r_i}^{(i)}(t) \end{bmatrix}.$$

Then $F(t) = \text{diagonal}\,\{\overline{F}_0(t), \overline{F}_1(t), \cdots, \overline{F}_{m-1}(t)\}$ and

$$G(t) = \begin{bmatrix} U_0(t) & V_0(t) \\ U_1(t) & V_1(t) \\ \vdots & \vdots \\ U_{m-1}(t) & V_{m-1}(t) \end{bmatrix} \equiv [\mathbf{U}(t) \quad \mathbf{V}(t)].$$

Let $n = \sum_{i=0}^{m-1} r_i$ and $r(t) = p(t) + q(t)$, then $F(t)$ and $G(t)$ are $n \times n$ and $n \times r(t)$ matrices, respectively. We shall denote the diagonal entries of $F(t)$ by $\{f_i(t)\}_{i=0}^{n-1}$ (for example, $f_0(t) = f_1(t) = \cdots = f_{r_0-1}(t) = \alpha_0(t)$). For the special examples considered in the previous section, the matrices $F(t)$, $G(t)$, and $J(t)$ are given by:

- *Scalar Carathéodory–Fejér*: $J(t) = \text{diagonal}\,\{1, -1\}$,

$$F(t) \equiv Z = \begin{bmatrix} 0 & & & \\ 1 & 0 & & \\ & \ddots & \ddots & \\ & & 1 & 0 \end{bmatrix},$$

$$G(t) = \begin{bmatrix} 1 & \beta_0(t) \\ 0 & \beta_1(t) \\ \vdots & \vdots \\ 0 & \beta_{n-1}(t) \end{bmatrix}. \tag{5}$$

- *Scalar Nevanlinna–Pick*: $J(t) = \text{diagonal}\,\{1, -1\}$,

$$F(t) = \begin{bmatrix} \alpha_0(t) & & & \\ & \alpha_1(t) & & \mathbf{0} \\ \mathbf{0} & & \ddots & \\ & & & \alpha_{n-1}(t) \end{bmatrix},$$

$$G(t) = \begin{bmatrix} 1 & \beta_0(t) \\ 1 & \beta_1(t) \\ \vdots & \vdots \\ 1 & \beta_{n-1}(t) \end{bmatrix}.$$

We shall show in the next sections that by applying a simple recursive procedure to $F(t)$ and $G(t)$ we obtain a cascade structure that satisfies the interpolation conditions (4) and, in fact, parameterizes all solutions. Meanwhile, we shall associate with the interpolation problem the time-variant displacement equation (a special case of a time-variant Stein equation)

$$R(t) - F(t)R(t-1)F^*(t) = G(t)J(t)G^*(t) \tag{6}$$

where the symbol * stands for Hermitian conjugation (complex conjugation for scalars), and the notation $F^*(t)$ (similarly $G^*(t)$, and other quantities throughout this paper) stands for $F(t)^*$ $(G(t)^*)$.

In this paper, we are interested in time-variant interpolation problems for which $R(t)$ is positive-definite for all t (in the sense described ahead by (8) in Theorem 3.1). This would then allow us to derive a recursive solution and a cascade of first-order lattice sections that satisfies the desired interpolation conditions. For this reason, we shall further assume that the interpolation data satisfy the following additional assumption that rules out degenerate cases and is automatically satisfied in many problems (this condition will become clearer throughout the presentation. See also [14], [16] for a related discussion in the time-invariant setting): if we define

$$\mathcal{U}(t) \equiv [\cdots \quad F(t)F(t-1)\mathbf{U}(t-2) \quad F(t)\mathbf{U}(t-1) \quad \mathbf{U}(t)]$$

we shall then require that $\mathcal{U}(t)$ satisfy the property

$$\mathcal{U}(t)\mathcal{U}^*(t) > \mu > 0 \quad \text{for all } t \tag{7}$$

where μ is a fixed constant. Consider, for example, the scalar Carathéodory–Fejér problem described earlier. For this case, we get $\mathcal{U}(t) = [\cdots\ \mathbf{0}\ \mathbf{0}\ \tilde{I}_n]$, where $\tilde{I}_n$ is the $n \times n$ reversed identity matrix. Hence, we always have $\mathcal{U}(t)\mathcal{U}^*(t) = I_n$, and (7) is automatically satisfied. The more general case that allows for positive-semidefinite matrices $R(t)$ is treated in [13], [17], [29]. We now prove the existence of solutions to Problem 2.1, where it is assumed that the interpolation data satisfy the stability and uniform boundedness conditions stated prior to Problem 2.1 and the nondegeneracy condition (7).

Theorem 3.1: Under the nondegeneracy condition (7), the tangential Hermite–Fejér problem is solvable if, and only if, there exists a real number $\epsilon > 0$, independent of t, such that the solution $R(t)$ of (6) satisfies

$$R(t) > \epsilon I \quad \text{for all } t \in \mathbb{Z}. \tag{8}$$

Proof: If $R(t)$ satisfies (8) and the interpolation data are stable and uniformly bounded, then the recursive algorithm described in the next sections finds a strictly contractive solution $\mathcal{S}$ to Problem 2.1. Conversely, assume there exists a strictly contractive upper-triangular operator $\mathcal{S}$ that solves (4). Then the following upper-triangular submatrix of $\mathcal{S}$ is also a strict contraction

$$\hat{\mathcal{S}}(t) = \begin{bmatrix} \ddots & & & \vdots \\ & S_{t-2,t-2} & S_{t-2,t-1} & S_{t-2,t} \\ & & S_{t-1,t-1} & S_{t-1,t} \\ \mathbf{0} & & & S_{tt} \end{bmatrix}.$$

If we further define

$$\mathcal{V}(t) = [\cdots \quad F(t)F(t-1)\mathbf{V}(t-2) \quad F(t)\mathbf{V}(t-1) \quad \mathbf{V}(t)]$$

it then follows from the stability and uniform boundedness of the interpolation data that $\mathcal{U}(t)$ and $\mathcal{V}(t)$ are bounded operators. Moreover, we can check by direct manipulations (see Appendix A) that because of (4) the following relation is satisfied

$$\mathcal{V}(t) = \mathcal{U}(t)\hat{\mathcal{S}}(t). \tag{9}$$

On the other hand, from (6) and the definition of $G(t)$, we see that $R(t) = \mathcal{U}(t)\mathcal{U}^*(t) - \mathcal{V}(t)\mathcal{V}^*(t) = \mathcal{U}(t)[I - \hat{\mathcal{S}}(t)\hat{\mathcal{S}}^*(t)]\mathcal{U}^*(t)$. But $\mathcal{U}(t)\mathcal{U}^*(t) > \mu$ (because of (7)) and $\hat{\mathcal{S}}(t)$ is a strict contraction, i.e., $I - \hat{\mathcal{S}}(t)\hat{\mathcal{S}}^*(t) \geq \bar{\epsilon} > 0$ for some real $\bar{\epsilon}$ independent of t. Hence, $R(t) > \epsilon I$ for every t and for some $\epsilon > 0$ (independent of t). ∎

A proof of Theorem 3.1 in full generality, and for block matrices, is described in [17]. Notice that the above proof explicitly uses the displacement equation (6) and shows (at least implicitly) that the contraction $\hat{\mathcal{S}}(t)$ can be constructed from $R(t)$ through (9). While an efficient construction will be given below, we remark that the proof establishes an important link between displacement equations and interpolation conditions. The displacement equation (6) imposes a so-called Toeplitz-like structure on $R(t)$; the matrix $G(t)$ in (6) is called a generator matrix [13], [16], [18]. We should stress at this point that we only know $F(t)$, $G(t)$, and $J(t)$, whereas the matrix $R(t) \equiv [r_{lj}(t)]_{l,j=0}^{n-1}$ itself is not known *a priori.* While there are approaches to the interpolation problem that give formulas involving $R^{-1}(t)$, the recursive solution described in the next sections does not need $R(t)$ or $R^{-1}(t)$; it only uses $F(t)$, $G(t)$, and $J(t)$.

IV. A Recursive Algorithm

We now focus on the time-variant displacement equation (6) and show that it allows the successive computation of the Schur complements of $R(t)$ to be reduced to a computationally efficient recursive procedure applied to the generator matrix $G(t)$. We shall later prove that this recursive algorithm leads to a cascade structure with the desired interpolation properties.

Let $R_i(t)$ denote the Schur complement of the leading $i \times i$ submatrix of $R(t)$. If $l_i(t)$ and $d_i(t)$ stand for the first column and the (0, 0) entry of $R_i(t)$, respectively, then (the positive-definiteness of $R(t)$ guarantees $d_i(t) > 0$ for all i)

$$R_i(t) - l_i(t)d_i^{-1}(t)l_i^*(t) = \begin{bmatrix} 0 & \mathbf{0} \\ \mathbf{0} & R_{i+1}(t) \end{bmatrix} \equiv \tilde{R}_{i+1}(t). \tag{10}$$

Hence, starting with an $n \times n$ matrix $R(t)$ and performing n consecutive Schur complement steps, we obtain the triangular factorization of $R(t)$, viz.,

$$\begin{aligned} R(t) &= l_0(t)d_0^{-1}(t)l_0^*(t) + \begin{bmatrix} 0 \\ l_1(t) \end{bmatrix} d_1^{-1}(t) \begin{bmatrix} 0 \\ l_1(t) \end{bmatrix}^* + \cdots \\ &= L(t)D^{-1}(t)L^*(t) \end{aligned}$$

where $D(t) = \text{diag}\{d_0(t), \cdots, d_{n-1}(t)\}$ ($D^{-1}(t)$ stands for $(D(t))^{-1}$), and the (nonzero parts of the) columns of the lower triangular matrix $L(t)$ are $\{l_0(t), \cdots, l_{n-1}(t)\}$. This procedure requires $O(n^3)$ operations (elementary additions and multiplications).

The point, however, is that the procedure can be speeded up to $O(r(t)n^2)$ operations in the case of matrices $R(t)$ that exhibit a time-variant displacement structure as in (6). In this case, the above (Gauss/Schur) reduction procedure can be shown to reduce to a recursion on the elements of the generator matrix $G(t)$. The computational advantage then follows from the fact that $G(t)$ has $r(t)n$ elements as compared to n^2 in $R(t)$. The following theorem shows that the triangular factor at time $(t-1)$, viz., $L(t-1)$ can be time-updated to $L(t)$ via a recursive procedure on $G(t)$.

Theorem 4.1: The Schur complements $R_i(t)$ are also structured with generator matrices $G_i(t)$, viz., $R_i(t) - F_i(t)R_i(t-1)F_i^*(t) = G_i(t)J(t)G_i^*(t)$, where $G_i(t)$ is an $(n-i) \times r(t)$ matrix that satisfies, along with $l_i(t)$, the following recursion: $G_0(t) = G(t)$, $F_0(t) = F(t)$

$$\begin{aligned} \begin{bmatrix} l_i(t) & \begin{matrix} \mathbf{0}_{1\times r(t)} \\ G_{i+1}(t) \end{matrix} \end{bmatrix} &= [F_i(t)l_i(t-1) \quad G_i(t)] \\ &\quad \cdot \begin{bmatrix} f_i^*(t) & h_i^*(t)J(t) \\ J(t)g_i^*(t) & J(t)k_i^*(t)J(t) \end{bmatrix} \end{aligned} \tag{11}$$

where $g_i(t)$ is the first row of $G_i(t)$, $F_i(t)$ is the $(n-i)\times(n-i)$ submatrix obtained after deleting the first row and column of $F_{i-1}(t)$, and $h_i(t)$ and $k_i(t)$ are arbitrary $r(t) \times 1$ and $r(t) \times r(t)$ matrices, respectively, chosen so as to satisfy the time-variant embedding relation

$$\begin{aligned} \begin{bmatrix} f_i(t) & g_i(t) \\ h_i(t) & k_i(t) \end{bmatrix} \begin{bmatrix} d_i(t-1) & \mathbf{0} \\ \mathbf{0} & J(t) \end{bmatrix} \begin{bmatrix} f_i(t) & g_i(t) \\ h_i(t) & k_i(t) \end{bmatrix}^* \\ = \begin{bmatrix} d_i(t) & \mathbf{0} \\ \mathbf{0} & J(t) \end{bmatrix} \end{aligned} \tag{12}$$

where $d_i(t)$ satisfies the time-update recursion

$$d_i(t) = f_i(t)d_i(t-1)f_i^*(t) + g_i(t)J(t)g_i^*(t).$$

Proof: We prove the result for $i = 0$. The same argument is valid for $i \geq 1$. Let $d_0(t)$, $l_0(t)$, and $g_0(t)$, denote the (0, 0) entry of $R(t)$, the first column of $R(t)$, and the first row of $G(t)$, respectively. It then follows from the displacement equation (6) that $l_0(t) = F(t)l_0(t-l)f_0^*(t) + G(t)J(t)g_0^*(t)$ and $d_0(t) = f_0(t)d_0(t-1)f_0^*(t) + g_0(t)J(t)g_0^*(t)$. Let $F_1(t)$ be the submatrix obtained after deleting the first row and column of $F(t)$. Using the expressions for $l_0(t)$, $d_0(t)$, and (10), it is straightforward to check that we can write

$$\begin{aligned}\tilde{R}_1(t) &- F(t)\tilde{R}_1(t-1)F^*(t)\\ &= -\frac{1}{d_0(t)}[f_0^*(t)F(t)l_0(t-1)g_0(t)J(t)G^*(t)\\ &+ G(t)J(t)g_0^*(t)l_0^*(t-1)F^*(t)f_0(t)\\ &- \frac{1}{d_0(t-1)}F(t)l_0(t-1)g_0(t)J(t)\\ &\cdot g_0^*(t)l_0^*(t-1)F^*(t)]\\ &+ G(t)J(t)\left\{J(t) - \frac{g_0^*(t)g_0(t)}{d_0(t)}\right\}J(t)G^*(t). \quad (13)\end{aligned}$$

We now verify that the right-hand side of the above expression can be put into the form of a perfect square by introducing some auxiliary quantities. Consider an $r(t)\times 1$ column vector $h_0(t)$ and an $r(t)\times r(t)$ matrix $k_0(t)$ that are defined to satisfy the following relations (in terms of the quantities that appear on the right-hand side of the above expression—explicit expressions for $h_0(t)$ and $k_0(t)$ will be given later)

$$h_0^*(t)J(t)h_0(t) = \frac{g_0(t)J(t)g_0^*(t)}{d_0(t)d_0(t-1)},$$

$$k_0^*(t)J(t)k_0(t) = J(t) - \frac{g_0^*(t)g_0(t)}{d_0(t)},$$

$$k_0^*(t)J(t)h_0(t) = -\frac{f_0(t)g_0^*(t)}{d_0(t)}. \quad (14)$$

Using $\{h_0(t), k_0(t)\}$, we can factor the right-hand side of (13) as $\tilde{G}_1(t)J(t)\tilde{G}_1^*(t)$, where $\tilde{G}_1(t) = F(t)l_0(t-1)h_0^*(t)J(t) + G(t)J(t)k_0^*(t)J(t)$. Recall that the first row and column of $\tilde{R}_1(t)$ are zero. Hence, the first row of $\tilde{G}_1(t)$ is zero, $\tilde{G}_1(t) = [\mathbf{0}\ G_1^{\mathrm{T}}(t)]^{\mathrm{T}}$. Moreover, it follows from (14) (and from the expression for $d_0(t)$) that

$$\begin{bmatrix} f_0(t) & g_0(t)\\ h_0(t) & k_0(t)\end{bmatrix}^*\begin{bmatrix} d_0^{-1}(t) & \mathbf{0}\\ \mathbf{0} & J(t)\end{bmatrix}\begin{bmatrix} f_0(t) & g_0(t)\\ h_0(t) & k_0(t)\end{bmatrix} = \begin{bmatrix} d_0^{-1}(t-1) & \mathbf{0}\\ \mathbf{0} & J(t)\end{bmatrix}$$

which is equivalent to (12) for $i = 0$. ∎

A. Discussion

Relation (11) is the general form of the generator recursion. It is expressed in terms of the quantities $h_i(t)$ and $k_i(t)$, which we still need to choose to satisfy the embedding relation (12). We shall show later that this is always possible. In fact, we shall derive explicit expressions for $h_i(t)$ and $k_i(t)$, which will allow us to substantially simplify the general recursion (11).

Before proceeding further, we first discuss the implications of the stability and uniform boundedness of the interpolation data $\{f_i(t), \mathbf{a}_i(t), \mathbf{b}_i(t)\}$ on the boundedness of the quantities $d_i(t)$ and $g_i(t)$ that are needed in the recursive procedure.

Lemma 4.1.1: The sequences $\{d_i(t)\}_{t\in\mathbf{Z}}$ and $\{g_i(t)\}_{t\in\mathbf{Z}}$ obtained through the recursive Schur reduction procedure are uniformly bounded. More specifically, there exist real numbers b_d, c_d, and c_g (independent of t) such that

$$0 < b_d < d_i(t) < c_d \quad\text{and}\quad \|g_i(t)\| < c_g \quad \text{for all } t. \quad (15)$$

Proof: It is clear that $\{d_0(t)\}_{t\in\mathbf{Z}}$ is uniformly bounded from above since $\{f_0(t)\}_{t\in\mathbf{Z}}$ is stable and $\{g_0(t)J(t)g_0^*(t)\}_{t\in\mathbf{Z}}$ is uniformly bounded. A similar argument shows that $\{l_0(t)\}_{t\in\mathbf{Z}}$ is also uniformly bounded. Now using (11) we get $g_1(t) = e_1F(t)l_0(t-1)h_0^*(t)J(t) + e_1G(t)J(t)k_0^*(t)J(t)$. But we shall show in Section VI that $\{h_0(t)\}_{t\in\mathbf{Z}}$ and $\{k_0(t)\}_{t\in\mathbf{Z}}$ can always be chosen to be uniformly bounded sequences. It then follows that $\{g_1(t)\}_{t\in\mathbf{Z}}$ is also uniformly bounded. Repeating this argument we conclude, by induction, that there exist real numbers $c_d > 0$ and $c_g > 0$ such that $d_i(t) < c_d$ and $\|g_i(t)\| < c_g$ for all $t\in\mathbb{Z}$.

To show that the sequence $\{d_i(t)\}_{t\in\mathbf{Z}}$ is also uniformly bounded from below, we use the fact that the successive Schur complements $R_i(t)$ also satisfy relations similar to (8). To see this, we rewrite each step of the Schur reduction procedure (10) in the form

$$R_i(t) = \begin{bmatrix} \frac{l_i(t)}{d_i(t)} & \mathbf{0}\\ & I_{n-i-1}\end{bmatrix}\begin{bmatrix} d_i(t) & \mathbf{0}\\ \mathbf{0} & R_{i+1}(t)\end{bmatrix} \cdot \begin{bmatrix} \frac{l_i(t)}{d_i(t)} & \mathbf{0}\\ & I_{n-i-1}\end{bmatrix}^* \quad (16)$$

which exhibits a congruence relation. We define, for notational simplicity,

$$A_i(t) \equiv \begin{bmatrix} \frac{l_i(t)}{d_i(t)} & \mathbf{0}\\ & I_{n-i-1}\end{bmatrix}$$

which is an invertible lower triangular matrix. Assume $R_i(t) > \epsilon_i I$ for some $\epsilon_i > 0$ independent of t ($\epsilon_0 = \epsilon$). Then clearly $d_i(t) > \epsilon_i$ and $A_i(t)$ is uniformly bounded. For any nonzero column vector $\mathbf{y}$, we can always write $\mathbf{y} = A_i^*(t)\mathbf{x}$ for some nonzero column vector $\mathbf{x}$, since $A_i(t)$ has full rank. Therefore,

$$\begin{aligned}\mathbf{y}^*\begin{bmatrix} d_i(t) & \mathbf{0}\\ \mathbf{0} & R_{i+1}(t)\end{bmatrix}\mathbf{y} &= \mathbf{x}^*A_i(t)\begin{bmatrix} d_i(t) & \mathbf{0}\\ \mathbf{0} & R_{i+1}(t)\end{bmatrix}A_i^*(t)\mathbf{x}\\ &= \mathbf{x}^*R_i(t)\mathbf{x} > \epsilon_i\|\mathbf{x}\|^2 = \epsilon_i\left\|A_i^{-*}(t)\mathbf{y}\right\|^2 \equiv \epsilon_{i+1}\|\mathbf{y}\|^2\end{aligned}$$

where in the last equality we used the fact that $\{A_i^{-1}(t)\}_{t\in\mathbf{Z}}$ is uniformly bounded. Consequently, $d_{i+1}(t) > \epsilon_{i+1}$ and we can choose $b_d = \min_{0\le i\le n-1}\epsilon_i$. ∎

Remark: Conversely, we can show that if $\{d_i(t)\}_{t\in\mathbf{Z}}$ is uniformly bounded from below, then (8) is satisfied. For this purpose, we apply the same argument and use (16) backwards starting with $R_{n-1}(t) = d_{n-1}(t)$ down to $R_0(t) = R(t)$.

V. Relation to the Interpolation Problem

The question now is: How does the recursive algorithm (11) relate to the Hermite–Fejér interpolation problem? The relevant fact to note here is that each recursive step gives rise to a linear first-order discrete-time system (in state-space form)

$$\begin{bmatrix} f_i^*(t) & h_i^*(t)J(t)\\ J(t)g_i^*(t) & J(t)k_i^*(t)J(t)\end{bmatrix}$$

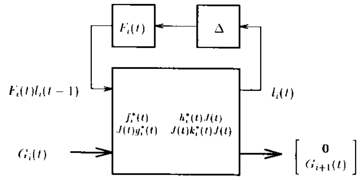


Fig. 1. One step of the time-variant generator recursion.

which appears on the right-hand side of (11). This can be thought of as the (state-space) transition matrix of a first-order system as

$$[\mathbf{x}_i(t+1) \quad \mathbf{y}_i(t)] = [\mathbf{x}_i(t) \quad \mathbf{w}_i(t)] \cdot \begin{bmatrix} f_i^*(t) & h_i^*(t)J(t) \\ J(t)g_i^*(t) & J(t)k_i^*(t)J(t) \end{bmatrix} \quad (17)$$

where $\mathbf{x}_i(t)$ denotes a (scalar) state, $\mathbf{w}_i(t)$ denotes a $1 \times r(t)$ row input vector, and $\mathbf{y}_i(t)$ denotes a $1 \times r(t)$ row output vector at time t. Moreover, the generator recursion (11) has a transmission-line picture in terms of a cascade of elementary steps as shown in Fig. 1, where each step depends on the parameters $\{f_i(t), g_i(t), h_i(t), k_i(t)\}$. The Δ block represents a storage element where the present value of $l_i(t)$ is stored for the next time instant.

The second important observation, which we shall verify very soon, is that each such section exhibits an intrinsic blocking property. The cascade of n sections would then exhibit certain global blocking properties, which will be shown to be equivalent to the desired interpolation conditions. Interesting enough, these blocking properties simply follow from the fact that each step of the Schur reduction procedure yields a matrix with a new zero row and column (as in (10)), which translates to a generator matrix with a new zero row (as in (11)).

A. Properties of the First-Order Sections

Let $\mathcal{T}_i = [T_{lj}^{(i)}]_{l,j=-\infty}^{\infty}$ denote the upper-triangular transfer operator associated with (17) (refer to (1)), where $T_{lj}^{(i)}$ denote the $(r(l) \times r(j))$ time-variant Markov parameters of $\mathcal{T}_i$ and are given by

$$\begin{aligned} T_{ll}^{(i)} &= J(l)k_i^*(l)J(l), \\ T_{l,l+1}^{(i)} &= J(l)g_i^*(l)h_i^*(l+1)J(l+1), \\ T_{lj}^{(i)} &= J(l)g_i^*(l)f_i^*(l+1)f_i^*(l+2)\cdots f_i^*(j-1) \\ &\quad \cdot h_i^*(j)J(j), \quad \text{for } j > l+1. \end{aligned}$$

The output and input sequences of $\mathcal{T}_i$ are clearly related by

$$\begin{bmatrix} \cdots & \mathbf{y}_i(-1) & \boxed{\mathbf{y}_i(0)} & \mathbf{y}_i(1) & \cdots \end{bmatrix} = \begin{bmatrix} \cdots & \mathbf{w}_i(-1) & \boxed{\mathbf{w}_i(0)} & \mathbf{w}_i(1) & \cdots \end{bmatrix} \mathcal{T}_i.$$

After n recursive steps (recall that $G(t)$ has n rows) we obtain a cascade of sections $\mathcal{T} = \mathcal{T}_0 \mathcal{T}_1 \cdots \mathcal{T}_{n-1}$. The cascade is depicted in Fig. 2, where we partitioned the input and output vectors of each section (namely $\mathbf{w}_i(t)$ and $\mathbf{y}_i(t)$) accordingly with $J(t) = J_{p(t)} \oplus -I_{q(t)}$: $\mathbf{w}_i(t) = [\mathbf{w}_{i,1}(t) \ \mathbf{w}_{i,2}(t)]$ and $\mathbf{y}_i(t) = [\mathbf{y}_{i,1}(t) \ \mathbf{y}_{i,2}(t)]$. The cascade may be regarded as a generalized transmission line.

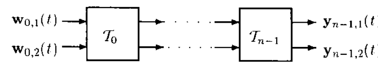


Fig. 2. Cascade of first-order time-variant sections.

Lemma 5.1.1: Each first-order section $\mathcal{T}_i$ is a bounded upper-triangular linear operator.

Proof: We already know that $\{f_i(t)\}_{t\in\mathbf{Z}}$ and $g_i(t)\}_{t\in\mathbf{Z}}$ are stable and uniformly bounded sequences, respectively. We shall show later in Section VI that $\{h_i(t), k_i(t)\}_{t\in\mathbf{Z}}$ can always be chosen to be uniformly bounded sequences as well. It is then a standard result that the (stability and) uniform boundedness of $\{f_i(t), g_i(t), h_i(t), k_i(t)\}$ assures the boundedness of the corresponding transfer operator $\mathcal{T}_i$ (see, e.g., [19]). ■

Moreover, if we define the direct sum $\mathcal{J} = \oplus_{t\in\mathbf{Z}} J(t)$, it then follows that each $\mathcal{T}_i$ also satisfies the following $\mathcal{J}$-losslessness property.

Lemma 5.1.2: Each first-order section $\mathcal{T}_i$ satisfies $\mathcal{T}_i \mathcal{J} \mathcal{T}_i^* = \mathcal{J}$ and $\mathcal{T}_i^* \mathcal{J} \mathcal{T}_i = \mathcal{J}$.

Proof: The proof is a direct consequence of the embedding construction (12), which leads to the relations

$$f_i^*(t)d_i^{-1}(t)f_i(t) + h_i^*(t)J(t)h_i(t) = d_i^{-1}(t-1),$$

$$f_i^*(t)d_i^{-1}(t)g_i(t) + h_i^*(t)J(t)k_i(t) = \mathbf{0},$$

$$g_i^*(t)d_i^{-1}(t)g_i(t) + k_i^*(t)J(t)k_i(t) = J(t).$$

Therefore, we can expand $d_i^{-1}(t)$ and write

$$\begin{aligned} d_i^{-1}(t) = {} & h_i^*(t+1)J(t+1)h_i(t+1) + f_i^*(t+1) \\ & \cdot h_i^*(t+2)J(t+2)h_i(t+2)f_i(t+1) \\ & + f_i^*(t+1)f_i^*(t+2)h_i^*(t+3)J(t+3) \\ & \cdot h_i(t+3)f_i(t+2)f_i(t+1) + \cdots . \end{aligned}$$

Now the tth element on the main diagonal of $\mathcal{T}_i \mathcal{J} \mathcal{T}_i^*$ (denoted by λ_{tt}) is given by

$$\begin{aligned} \lambda_{tt} = {} & J(t)[k_i^*(t)J(t)k_i(t) + g_i^*(t)h_i^*(t+1)J(t+1) \\ & \cdot h_i(t+1)g_i(t) + g_i^*(t)f_i^*(t+1)h_i^*(t+2) \\ & \cdot J(t+2)h_i(t+2)f_i(t+1)g_i(t) + \cdots]J(t). \end{aligned}$$

Using the expression for $d_i^{-1}(t)$ we obtain $\lambda_{tt} = J(t) - J(t)g_i^*(t)[d_i^{-1}(t) - d_i^{-1}(t)]g_i(t)J(t) = J(t)$. The same argument can be used to show that the off-diagonal elements of $\mathcal{T}_i \mathcal{J} \mathcal{T}_i^*$ are zero. We use a similar procedure for proving that $\mathcal{T}_i^* \mathcal{J} \mathcal{T}_i = \mathcal{J}$. ■

Furthermore, each section $\mathcal{T}_i$ satisfies an important blocking property in the following sense (using definition (2)).

Theorem 5.1.1: Each first-order section $\mathcal{T}_i$ satisfies

$$[\cdots \quad f_i(t)f_i(t-1)g_i(t-2) \quad f_i(t) \cdot g_i(t-1) \quad g_i(t) \quad ?]\mathcal{T}_i = [\mathbf{0} \quad ?]$$

where $g_i(t)$ is at the tth position of the row vector. Consequently, $g_i(t) \bullet \mathcal{T}_i(f_i(t)) = \mathbf{0}$.

Proof: This follows directly from the embedding result (12) (as well as from the fact that each step of the generator recursion (11) produces a new zero row). The output of $\mathcal{T}_i$ at time t is given by

$$\begin{aligned} y_i(t) &= \cdots + f_i(t)f_i(t-1)g_i(t-2)T_{t-2,\,t} \\ &\quad + f_i(t)g_i(t-1)T_{t-1,\,t} + g_i(t)T_{tt} \\ &= [-d_i(t-1) + d_i(t-1)]f_i(t)h_i^*(t)J(t) = \mathbf{0} \end{aligned}$$

where we substituted the expressions for the Markov parameters $\{T_{jt}\}_{j\le t}$ and used

$$\begin{aligned} d_i(t) &= g_i(t)J(t)g_i^*(t) + f_i(t)g_i(t-1)J(t-1) \\ &\quad \cdot g_i^*(t-1)f_i^*(t) \\ &\quad + f_i(t)f_i(t-1)g_i(t-2)J(t-2) \\ &\quad \cdot g_i^*(t-2)f_i^*(t-1)f_i^*(t) + \cdots . \end{aligned}$$

The same argument holds for the previous outputs. ∎

In general terms, the blocking property means that when $g_i(t)$ (which is the first row of $G_i(t)$) is applied to $\mathcal{T}_i$ we obtain a zero output at $f_i(t)$ at time t. We say that $f_i(t)$ is a time-variant transmission-zero of $\mathcal{T}_i$ and $g_i(t)$ is the associated time-variant left-zero direction. We remark that the concepts of transmission zeros and blocking directions are central to many problems in network theory and linear systems [20].

Remark: In the time-invariant case, the above blocking property reduces to (dropping the time index)

$$[\cdots \quad f_i^2 g_i \quad f_i g_i \quad g_i \quad ?]\mathcal{T}_i = [\mathbf{0} \quad ?]$$

where $\mathcal{T}_i$ is now an upper-triangular Toeplitz operator ($T_{lj}^{(i)} = T_{|l-j|}^{(i)}$). Hence, $g_iT_0^{(i)} + g_iT_1^{(i)}f_i + g_iT_2^{(i)}f_i^2 + \cdots = \mathbf{0}$, or equivalently, $g_iT_i(f_i) = \mathbf{0}$, where $T_i(z)$ denotes the transfer matrix of the time-invariant discrete-time system, $T_i(z) = Jk_i^*J + Jg_i^*[z^{-1} - f_i^*]^{-1}h_i^*J$.

B. Properties of the Cascade

The $\mathcal{J}$-losslessness and blocking properties of each section $\mathcal{T}_i$ reflect on the entire cascade $\mathcal{T}$. The following result is a direct consequence of Lemmas 5.1.1 and 5.1.2 and the definition of $\mathcal{T}$.

Lemma 5.2.1: The cascade $\mathcal{T}$ is a bounded upper-triangular linear operator and satisfies $\mathcal{T}\mathcal{J}\mathcal{T}^* = \mathcal{T}^*\mathcal{J}\mathcal{T} = \mathcal{J}$.

It also follows from Theorem 5.1.1 that $\mathcal{T}$ satisfies a global blocking property.

Theorem 5.2.1: The entire cascade $\mathcal{T}$ satisfies the global blocking property

$$\begin{aligned} [\cdots \quad F(t)F(t-1)G(t-2) \quad & F(t)G(t-1) \\ & G(t) \quad \mathbf{0} \quad \mathbf{0} \quad \cdots]\mathcal{T} = [\mathbf{0} \quad ?] \end{aligned} \tag{18}$$

where $G(t)$ is in the tth position. That is, if we apply to $\mathcal{T}$ the block input

$$\begin{aligned} \tilde{\mathbf{U}}(t) = [\cdots \quad F(t)F(t-1)G(t-2) \quad & F(t) \\ \cdot G(t-1) \quad & G(t) \quad \mathbf{0} \quad \mathbf{0} \quad \cdots] \end{aligned}$$

then the output is zero up to and including time t.

Proof: This follows from the generator recursion (11) and from the Jordan structure of $F(t)$. When the first row of $\tilde{\mathbf{U}}(t)$ goes through the first section $\mathcal{T}_0$, it annihilates the output of the entire cascade $\mathcal{T}$ due to the blocking property of $\mathcal{T}_0$. When the second row of $\tilde{\mathbf{U}}(t)$ goes through $\mathcal{T}_0$, we obtain at the output of $\mathcal{T}_0$ (as a consequence of (11) and the Jordan structure of $F(t)$, and similar to the proof of Theorem 5.1.1) a zero-direction vector for $\mathcal{T}_1$, which again annihilates the output of the entire cascade $\mathcal{T}$, and so on. ∎

Expression (18) is closely related to the interpolation conditions of Problem 2.1. To clarify this, we denote by $s_i = \sum_{p=0}^{i-1} r_p$, $s_0 = 0$, the total size of the Jordan blocks prior to the ith Jordan block, $\overline{F}_i(t)$. By comparing terms on both sides of (18) (and by using (3) and the Jordan structure of $F(t)$) we can verify that (18) is equivalent to the following result.

Theorem 5.2.2: The entire cascade $\mathcal{T}$ satisfies

$$\begin{aligned} [e_{s_i}G(t) \quad e_{s_i+1}G(t) \quad \cdots \quad & e_{s_i+r_i-1}G(t)] \\ & \bullet \mathcal{H}_{\mathcal{T}}^{r_i}(\alpha_i(t)) = \mathbf{0}. \end{aligned} \tag{19}$$

The row vector on the left-hand side of (19) is composed of the r_i row vectors in $[U_i(t) \;\; V_i(t)]$ associated with $\alpha_i(t)$, viz.,

$$\left[u_1^{(i)}(t) \quad v_1^{(i)}(t) \quad u_2^{(i)}(t) \quad v_2^{(i)}(t) \quad \cdots \quad u_{r_i}^{(i)}(t) \quad v_{r_i}^{(i)}(t)\right].$$

The Jordan structure of $F(t)$ is essential in deriving (19) (see [16, Chapter 9] for more motivation and detailed calculations).

C. From Blocking Properties to Interpolation Properties

We now verify that the global blocking property (18) (or equivalently, (19)) is equivalent to the desired interpolation properties stated in Problem 2.1.

For this purpose, recall that each first-order transfer operator $\mathcal{T}_i$ is formed of $r(l)\times r(j)$ matrix entries $T_{lj}^{(i)}$. We thus partition $T_{lj}^{(i)}$ accordingly with $J(l)$ and $J(j)$, viz.,

$$T_{lj}^{(i)} = \begin{bmatrix} T_{11,\,i}^{lj} & T_{12,\,i}^{lj} \\ T_{21,\,i}^{lj} & T_{22,\,i}^{lj} \end{bmatrix}$$

where $T_{11,\,i}^{lj}$, $T_{12,\,i}^{lj}$, $T_{21,\,i}^{lj}$, and $T_{22,\,i}^{lj}$ are $p(l)\times p(j)$, $p(l)\times q(j)$, $q(l)\times p(j)$, and $q(l)\times q(j)$ matrices, respectively. We further define the upper-triangular operators

$$\mathcal{T}_{12}^{(i)} = [T_{12,\,i}^{lj}]_{l,\,j=-\infty}^{\infty} \quad \text{and} \quad \mathcal{T}_{22}^{(i)} = [T_{22,\,i}^{lj}]_{l,\,j=-\infty}^{\infty}.$$

Lemma 5.3.1: The operator $\mathcal{T}_{12}^{(i)}\mathcal{T}_{22}^{-(i)}$ is upper-triangular and strictly contractive.

Proof: See Appendix B. ∎

We also partition the matrix entries T_{lj} of the cascade $\mathcal{T}$ accordingly with $J(l)$ and $J(j)$

$$T_{lj} = \begin{bmatrix} T_{11}^{lj} & T_{12}^{lj} \\ T_{21}^{lj} & T_{22}^{lj} \end{bmatrix}$$

and consider the triangular operators

$$\mathcal{T}_{11} = [T_{11}^{lj}]_{l,\,j=-\infty}^{\infty}, \qquad \mathcal{T}_{21} = [T_{21}^{lj}]_{l,\,j=-\infty}^{\infty},$$

$$\mathcal{T}_{12} = [T_{12}^{lj}]_{l,\,j=-\infty}^{\infty}, \qquad \mathcal{T}_{22} = [T_{22}^{lj}]_{l,\,j=-\infty}^{\infty}.$$

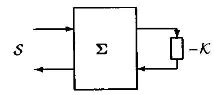


Fig. 3. Scattering interpretation.

We can also verify (as done in Appendix B for the proof of Lemma 5.3.1 and by using Theorem 5.4.1 ahead) that $\mathcal{S} \equiv -\mathcal{T}_{12}\mathcal{T}_{22}^{-1}$ is also an upper-triangular strictly contractive operator. Moreover, it follows from Theorem 5.2.2 that $\mathcal{S}$ satisfies the required interpolation conditions. For example, we conclude from the blocking property of $\mathcal{T}$ that

$$[\cdots \quad f_0(t)f_0(t-1)g_0(t-2) \quad f_0(t) \cdot g_0(t-1) \quad g_0(t) \quad \mathbf{0} \quad \mathbf{0} \quad \cdots]\begin{bmatrix}\mathcal{T}_{12}\\ \mathcal{T}_{22}\end{bmatrix} = [\mathbf{0} \quad ?]$$

or equivalently, $v_1^{(0)}(t) = u_1^{(0)}(t) \bullet \mathcal{S}(\alpha_0(t))$. This argument can be extended to show that $\mathcal{S}$ satisfies the remaining interpolation conditions. In fact, all solutions to Problem 2.1 can be parameterized in terms of a linear fractional transformation based on $\mathcal{T}$ as follows.

Theorem 5.3.1: All solutions $\mathcal{S}$ of the tangential Hermite–Fejér problem are given through a linear fractional transformation of a strictly contractive upper-triangular operator $\mathcal{K}$

$$\mathcal{S} = -[\mathcal{T}_{11}\mathcal{K} + \mathcal{T}_{12}][\mathcal{T}_{21}\mathcal{K} + \mathcal{T}_{22}]^{-1}. \tag{20}$$

Proof: The proof of this fact is by now a standard one and similar to the time-invariant counterpart (see, e.g., [8], [10], [11], [17] for details). ■

The solution $\mathcal{S}$ in (20) has a nice scattering interpretation. Recall that the input and output row vectors of $\mathcal{T}$ are related (schematically) by

$$[\mathbf{y}_{n-1,1}(t) \quad \mathbf{y}_{n-1,2}(t)] = [\mathbf{w}_{0,1}(t) \quad \mathbf{w}_{0,2}(t)]\mathcal{T}.$$

The scattering operator $\mathbf{\Sigma}$ associated with $\mathcal{T}$ is defined by the relation

$$[\mathbf{y}_{n-1,1}(t) \quad \mathbf{w}_{0,2}(t)] = [\mathbf{w}_{0,1}(t) \quad \mathbf{y}_{n-1,2}(t)]\mathbf{\Sigma}.$$

In this setting, the solution $\mathcal{S}$ is the transfer operator from the top left $(1 \times p(t))$ input to the bottom left $(1 \times q(t))$ output, with a strictly contractive load $(-\mathcal{K})$ at the right end, as shown in Fig. 3.

In summary, we are led to the following recursive procedure that parameterizes all solutions to the Hermite–Fejér problem (this general procedure will be greatly simplified in the next section—see Algorithm 6.2.1 ahead).

Algorithm 5.3.1: The Hermite–Fejér Problem 2.1 can be recursively solved as follows:

- Construct $F(t)$, $G(t)$, and $J(t)$ from the interpolation data as described in Section III.
- Start with $G(t)$ and apply n steps of the generator recursion (11). This leads to a cascade of first-order sections that are completely determined by the $\{f_i(t), g_i(t), h_i(t), k_i(t)\}$ as in (17).
- The entire transfer-operator parameterizes all solutions in terms of an arbitrary strictly contractive upper-triangular operator $\mathcal{K}$ as shown in (20). Specifically, any such $\mathcal{S}$ satisfies the interpolation conditions $\mathbf{a}_i(t) \bullet \mathcal{H}_{\mathcal{S}}^{r_i}(\alpha_i(t)) = \mathbf{b}_i(t)$ for $0 \le i \le m-1$ and $t \in \mathbb{Z}$.

D. A Global State-Space Description

Each first-order section $\mathcal{T}_i$ has a state-space description in terms of the quantities $f_i(t)$, $g_i(t)$, $h_i(t)$, and $k_i(t)$, as shown in (17). We also include here, for completeness, a state-space description for the entire cascade $\mathcal{T}$. Following an argument similar to that presented in [14], [16], [22] and in Appendix C, we can derive the following result.

Theorem 5.4.1: The entire cascade $\mathcal{T}$ admits an n-dimensional time-variant state-space description of the form

$$[\mathbf{x}(t+1) \quad \mathbf{y}(t)] = [\mathbf{x}(t) \quad \mathbf{w}(t)]\begin{bmatrix} F^*(t) & H^*(t)J(t) \\ J(t)G^*(t) & J(t)K^*(t)J(t)\end{bmatrix}$$

where $H(t)$ and $K(t)$ are $r(t) \times n$ and $r(t) \times r(t)$ matrices that satisfy the time-variant embedding relation

$$\begin{bmatrix} F(t) & G(t) \\ H(t) & K(t)\end{bmatrix}\begin{bmatrix} R(t-1) & \mathbf{0} \\ \mathbf{0} & J(t)\end{bmatrix}\begin{bmatrix} F(t) & G(t) \\ H(t) & K(t)\end{bmatrix}^* = \begin{bmatrix} R(t) & \mathbf{0} \\ \mathbf{0} & J(t)\end{bmatrix}.$$

Furthermore, $H(t)$ and $K(t)$ can be expressed in terms of $F(t)$, $G(t)$, and the Cholesky factor $\overline{L}(t)$ $(R(t) = \overline{L}(t)\overline{L}^*(t))$ as

$$H(t) = \mathbf{\Theta}^{-1}(t)J(t)G^*(t)[\overline{L}^*(t) - \tau(t)\overline{L}^*(t-1) \cdot F^*(t)]^{-1}[\tau(t)\overline{L}^{-1}(t-1) - \overline{L}^{-1}(t)F(t)]$$

$$K(t) = \mathbf{\Theta}^{-1}(t)[I_{r(t)} - J(t)G^*(t)[\overline{L}^*(t) - \tau(t) \cdot \overline{L}^*(t-1)F^*(t)]^{-1}\overline{L}^{-1}(t)G(t)]$$

where $\mathbf{\Theta}(t)$ is a $J(t)$-unitary matrix and $\tau(t)$ is a scalar on the unit circle.

We remark that the above formulas are time-variant analogues of expressions obtained in [12] and [22]. They are also related to the global solution presented in [11].

VI. Simplified Generator Recursion: Lattice Structures

The theory developed so far gives a recursive solution of the Hermite–Fejér Problem 2.1, and one could in principle stop here. The point, however, is that the generator recursion can still be greatly simplified. We shall pursue this line of argument here and show that the above cascade can be reduced to a simpler form by conveniently choosing the free parameters $h_i(t)$ and $k_i(t)$. Recall that these parameters are to be chosen at will as long as they are uniformly bounded (over t) and the embedding relation (12) is satisfied. We shall show in this section that a simple convenient choice is possible that will reduce the first-order sections to simple (so-called lattice) forms. To begin with, we show in Appendices C and D the following result (see [22] for the time-invariant results).

Lemma 6.1: It follows from the embedding relation (12) that the parameters $h_i(t)$ and $k_i(t)$ can be expressed in terms of the known quantities $f_i(t)$, $g_i(t)$, and $d_i(t)$ as

$$h_i(t) = \Theta_i^{-1}(t)\left\{\frac{1-\tau_i^*(t)f_i(t)}{\tau_i^*(t)d_i(t)-d_i(t-1)f_i^*(t)}J(t)g_i^*(t)\right\}$$

$$k_i(t) = \Theta_i^{-1}(t)\left\{I_{r(t)} - \frac{\tau_i^*(t)J(t)g_i^*(t)g_i(t)}{\tau_i^*(t)d_i(t)-d_i(t-1)f_i^*(t)}\right\} \quad (21)$$

where $\Theta_i(t)$ is an arbitrary $J(t)$-unitary matrix, $(\Theta_i(t)J(t)\Theta_i^*(t) = J(t))$, and $\tau_i(t)$ is an arbitrary complex number chosen on the circle $|\tau_i(t)|^2 = d_i(t-1)/d_i(t)$. Moreover, choosing $\Theta_i(t) = I_{r(t)}$ and $\tau_i(t)$ in the opposite direction of $f_i(t)$, then $\{h_i(t), k_i(t)\}_{t\in\mathbf{Z}}$ are guaranteed to be uniformly bounded.

A. Proper Generator Form

In fact, alternative and computationally more attractive choices for $\Theta_i(t)$ (other than $\Theta_i(t) = I$) are also possible as we now further elaborate. Recall that $g_i(t)$ denotes the first row of $G_i(t)$. If $g_i(t)$ has positive $J(t)$–norm, i.e., $g_i(t)J(t)g_i^*(t) > 0$, then we can always reduce it to the form

$$g_i(t)\Theta_i(t) = [\delta_i(t) \quad 0 \quad \cdots \quad 0] \quad (22)$$

where the (single) nonzero entry $\delta_i(t)$ is in the first position. For this purpose we can, for example, implement $\Theta_i(t)$ as a sequence of elementary rotations, as a Householder transformation, or in other convenient ways. If, on the other hand, $g_i(t)$ has negative $J(t)$-norm, i.e., $g_i(t)J(t)g_i^*(t) < 0$, then we can always reduce it to the form

$$g_i(t)\Theta_i(t) = [0 \quad \cdots \quad 0 \quad \delta_i(t)] \quad (23)$$

where the (single) nonzero entry $\delta_i(t)$ is in the last position. In either case we say that $G_i(t)$ is reduced to proper form. The point to check is whether the corresponding $\{h_i(t), k_i(t)\}_{t\in\mathbf{Z}}$ in (21) will still be uniformly bounded.

To guarantee this we add the additional assumption that the sequence $\{g_i(t)J(t)g_i^*(t)\}_{t\in\mathbf{Z}}$ be uniformly bounded from below (it is clearly uniformly bounded from above because of (15)). That is, we assume that there exists a real number $b_g > 0$ such that $b_g < |g_i(t)J(t)g_i^*(t)|$ for all t. Observe that this assumption is automatically satisfied for strictly lower triangular matrices $F(t)$ (as in the Carathéodory–Fejér case, for example), since in these cases we have $g_i(t)J(t)g_i^*(t) = d_i(t)$ and $\{d_i(t)\}_{t\in\mathbf{Z}}$ is uniformly bounded from below as established in Lemma 4.1.1.

Lemma 6.1.1: Under the assumption that $\{g_i(t)J(t)g_i^*(t)\}_{t\in\mathbf{Z}}$ is uniformly bounded from below, the sequences $\{h_i(t), k_i(t)\}_{t\in\mathbf{Z}}$ in (21) that are obtained by using the $\Theta_i(t)$ that reduces $G_i(t)$ to proper form are uniformly bounded.

Proof: It follows from the boundedness of $\{g_i(t)J(t)g_i^*(t)\}_{t\in\mathbf{Z}}$ that $\{\delta_i(t)\}_{t\in\mathbf{Z}}$ is also uniformly bounded from below since $|\delta_i(t)|^2 = |g_i(t)J(t)g_i^*(t)|$. But $\|g_i(t)\Theta_i(t)\| = |\delta_i(t)| \le \|g_i(t)\|\,\|\Theta_i(t)\|$ and hence, $\|\Theta_i^{-1}(t)\| \le \|g_i(t)\|/|\delta_i(t)|$. We thus conclude that $\{\Theta_i^{-1}(t)\}_{t\in\mathbf{Z}}$ is uniformly bounded. It then follows, as argued in Appendix D, that the resulting sequences $\{h_i(t), k_i(t)\}_{t\in\mathbf{Z}}$ are uniformly bounded. ∎

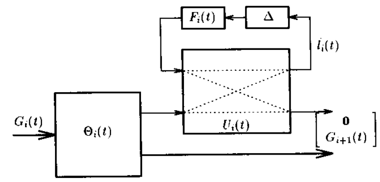


Fig. 4. A positive proper step of the generator recursion.

B. Simplified Recursions

The generator recursion is greatly simplified if we choose $\Theta_i(t)$ at each step so as to reduce the generator to proper form. For example, in the case $g_i(t)J(t)g_i^*(t) > 0$, (11) reduces to

$$\begin{bmatrix} l_i(t) & \begin{matrix} \mathbf{0} \\ G_{i+1}(t) \end{matrix} \end{bmatrix} = [F_i(t)l_i(t-1) \quad G_i(t)] \cdot \begin{bmatrix} f_i^*(t) & \frac{\phi_i(t)\delta_i(t)}{d_i(t-1)}[1 \quad \mathbf{0}] \\ \Theta_i(t)\begin{bmatrix}\delta_i(t) \\ \mathbf{0}\end{bmatrix} & \Theta_i(t)\begin{bmatrix} -\phi_i(t)f_i(t) & \mathbf{0} \\ \mathbf{0} & I \end{bmatrix} \end{bmatrix} \quad (24)$$

where we defined the scalar quantity

$$\phi_i(t) = \frac{1-\tau_i(t)f_i^*(t)}{1-\tau_i^*(t)f_i(t)}\tau_i^*(t).$$

The above generator expression has a simple array interpretation. It shows that $G_{i+1}(t)$ can be obtained as follows: multiply $G_i(t)$ by $\Theta_i(t)$ and keep the last $(r(t)-1)$ columns; the first column of $G_{i+1}(t)$ is obtained as a linear combination of $F_i(t)l_i(t-1)$ and the first column of $G_i(t)\Theta_i(t)$. In fact, this linear combination is obtained through an elementary unitary transformation. If we let $\overline{x}_i(t)$ and $x_{i+1}(t)$ denote the first columns of $G_i(t)\Theta_i(t)$ and $G_{i+1}(t)$, respectively, and define the normalized column $\overline{l}_i(t) = l_i(t)d_i^{-1/2}(t)$, then using the above generator recursion we have (this also follows from (11))

$$\begin{bmatrix} \overline{l}_i(t) & \begin{matrix} 0 \\ x_{i+1}(t) \end{matrix} \end{bmatrix} = [F_i(t)\overline{l}_i(t-1) \quad \overline{x}_i(t)]U_i(t) \quad (25)$$

where $U_i(t)$ is the 2×2 unitary matrix $(U_i(t)U_i^*(t) = I_2)$ given by

$$U_i(t) = \begin{bmatrix} f_i^*(t) & \rho_i(t) \\ \rho_i^*(t) & -f_i(t) \end{bmatrix}\begin{bmatrix} |\tau_i(t)| & 0 \\ 0 & \phi_i(t) \end{bmatrix},$$

$$\rho_i(t) = \frac{\delta_i(t)}{\sqrt{d_i(t-1)}}.$$

This is summarized in Fig. 4 where we show the structure of a single step: the first column of $G_i(t)$ goes through the top line and the last $(r(t)-1)$ columns propagate through the bottom line. The output $\overline{x}_i(t)$ of the top line (which is the first column of $G_i(t)\Theta_i(t)$) goes through an elementary unitary rotation $U_i(t)$, along with $F_i(t)\overline{l}_i(t-1)$, and generates the first input of the next section $(x_{i+1}(t))$, as well as $\overline{l}_i(t)$.

A similar argument holds for the case $g_i(t)J(t)g_i^*(t) < 0$ and leads to Fig. 5. Now, however, the elementary unitary transformation $U_i(t)$ is replaced by an elementary hyperbolic transformation $V_i(t)$

$$V_i(t)\begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}V_i^*(t) = \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}.$$

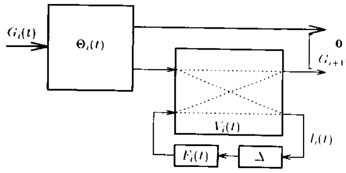


Fig. 5. A negative proper step of the generator recursion.

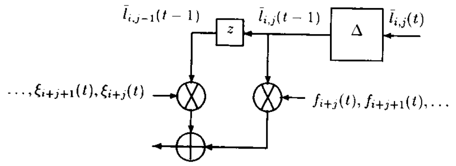


Fig. 6. First-order time-variant tapped-delay line.

Let $\bar{y}_i(t)$ and $y_{i+1}(t)$ denote the last columns of $G_i(t)\Theta_i(t)$ and $G_{i+1}(t)$, respectively. The generator recursion (11) then reduces to

$$\begin{bmatrix} l_i(t) & \begin{matrix} \mathbf{0} \\ G_{i+1}(t) \end{matrix} \end{bmatrix} = [F_i(t)l_i(t-1) \quad G_i(t)] \cdot \begin{bmatrix} f_i^*(t) & \frac{\phi_i(t)\delta_i(t)}{d_i(t-1)}[\mathbf{0} \quad 1] \\ \Theta_i(t)\begin{bmatrix} \mathbf{0} \\ -\delta_i(t) \end{bmatrix} & \Theta_i(t)\begin{bmatrix} I & \mathbf{0} \\ \mathbf{0} & -\phi_i(t)f_i(t) \end{bmatrix} \end{bmatrix} \tag{26}$$

which has the following array interpretation: multiply $G_i(t)$ by $\Theta_i(t)$ and keep the first $(r(t)-1)$ columns; the last column of $G_{i+1}(t)$ is obtained through the elementary transformation

$$\begin{bmatrix} \bar{l}_i(t) & \begin{matrix} 0 \\ y_{i+1}(t) \end{matrix} \end{bmatrix} = [F_i(t)\bar{l}_i(t-1) \quad \bar{y}_i(t)]V_i(t)$$

where

$$V_i(t) = \begin{bmatrix} f_i^*(t) & \rho_i(t) \\ -\rho_i^*(t) & -f_i(t) \end{bmatrix} \begin{bmatrix} |\tau_i(t)| & 0 \\ 0 & \phi_i(t) \end{bmatrix}.$$

We are thus led to the following simplified version of Algorithm 5.3.1.

Algorithm 6.2.1: Make the additional assumption that the $\{g_i(t)J(t)g_i^*(t)\}_{t\in\mathbf{Z}}$ are uniformly bounded from below. Then the Hermite–Fejér Problem 2.1 can be recursively solved as follows: construct $F(t)$, $G(t)$, and $J(t)$ from the interpolation data as described in Section III. Start with $G_0(t) = G(t)$, $F_0(t) = F(t)$, and repeat for $i = 0, 1, \cdots, n-1$.

- At step i we have $F_i(t)$ and $G_i(t)$.
- Verify whether the first row of $G_i(t)$ has positive or negative $J(t)$ norm, and use a convenient $J(t)$-unitary rotation $\Theta_i(t)$ that would reduce $g_i(t)$ to the corresponding proper form ((22) or (23)). This determines $\delta_i(t)$ and $\rho_i(t)$.
- Choose a scalar $\tau_i(t)$ on the circle $|\tau_i(t)|^2 = d_i(t-1)/d_i(t)$ and in the opposite direction of $f_i(t)$. This determines $\phi_i(t)$.
- Apply either of the simplified recursions (24) or (26), depending on whether $g_i(t)J(t)g_i^*(t)$ is positive or negative. Each such step completely characterizes a first-order section in simplified form, as it appears on the right-hand sides of (24) and (26).

C. Triangular Array Implementation

We now discuss in more detail the computational aspects of the above algorithm. More specifically, we show that it can be implemented as a triangular array of processing elements, where each cell consists of a rotation matrix followed by a storage element and a tapped-delay filter.

To begin with, the matrix-vector product $F_i(t)\bar{l}_i(t-1)$, indicated in the feedback line in either Figs. 4 or 5, can be implemented by a time-variant tapped-delay filter. To clarify this, recall that $F_i(t)$ has a Jordan structure of the form

$$F_i(t) = \begin{bmatrix} f_i(t) & & & & \\ \xi_{i+1}(t) & f_{i+1}(t) & & \mathbf{0} & \\ & \xi_{i+2}(t) & f_{i+2}(t) & & \\ \mathbf{0} & & \xi_{i+3}(t) & f_{i+3}(t) & \\ \vdots & & & & \ddots \end{bmatrix}, \quad \xi_j(t) = 1 \text{ or } 0.$$

That is, $F_i(t)$ has ones and zeros on the first subdiagonal (whose entries we shall denote by $\xi_j(t)$, $j > i$, $\xi_j(t) = 1$ or 0, ($\xi_i(t) = 0$)). The entries on the first subdiagonal have zeros at the starting points of Jordan blocks. If we denote the entires of $\bar{l}_i(t-1)$ by $\bar{l}_i(t-1) = [\bar{l}_{i,0}(t-1) \ \bar{l}_{i,1}(t-1) \ \cdots]^{\mathrm{T}}$, then the computation of the elements of the column vector $F_i(t)\bar{l}_i(t-1)$ involves operations of the form

$$f_{i+j}(t)\bar{l}_{i,j}(t-1) + \xi_{i+j}(t)\bar{l}_{i,j-1}(t-1), \qquad j \geq 0.$$

Hence, the entries of $F_i(t)\bar{l}_i(t-1)$ can be obtained as outputs of time-variant tapped-delay filters, whose coefficients are given by the rows of $F_i(t)$. This is shown in Fig. 6. The Δ block stores the elements of $\bar{l}_i(t)$ for the next time instant, and multiplication by $F_i(t)$ corresponds to a first-order time-variant finite-impulse-response filter whose coefficients vary (for a fixed t) as follows: when the first row of $G_i(t)$ is fed through $\Theta_i(t)$, the filter coefficients are $f_i(t)$ and 0. When the $(j+1)$th row of $G_i(t)$ is fed in, the filter coefficients are $\xi_{i+j}(t)$ and $f_{i+j}(t)$, and so on. More precisely, recall that $G_i(t)$ has $(n-i)$ rows. Hence, we can decompose Fig. 4 (or Fig. 5) into $(n-i)$ elementary sections as shown in Fig. 7 for the positive case. Each section consists of the same $J(t)$-unitary transformation $\Theta_i(t)$, followed by an elementary rotation $U_i(t)$, the storage element Δ, and the tapped-delay filter whose coefficients vary from section to section. One row of $G_i(t)$ is applied to each section. The outputs of the layer are then the rows of $G_{i+1}(t)$. Notice that the output of the top section is zero, since each generator step produces one zero row.

A similar layer also exists for each negative step ($g_i(t)J(t)g_i^*(t) < 0$), except that now each section has a slightly different structure where the storage element and the tapped-delay filter act on the bottom line, as shown in Fig. 5. If we represent each section in either Figs. 4 or 5 by a square box, then n steps of the generator recursion correspond to a triangular array as depicted in Fig. 8. The first layer of the array operates on the n rows of $G(t)$ and produces the $n-1$ rows of $G_1(t)$. The second layer operates on the rows of $G_1(t)$ and produces the $n-2$ rows of $G_2(t)$, and so on. It

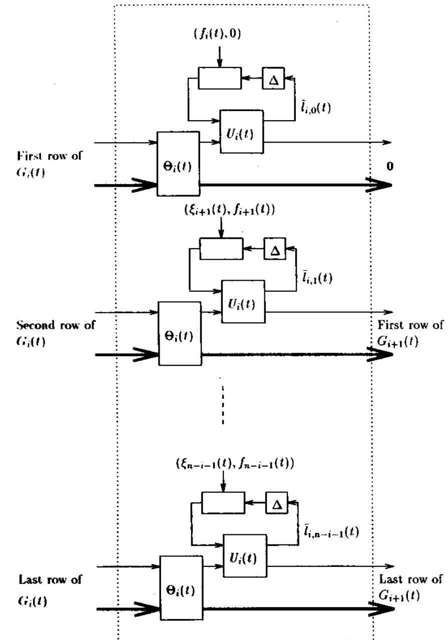


Fig. 7. A layer of elementary sections: $g_i(t)J(t)g_i^*(t) > 0$.

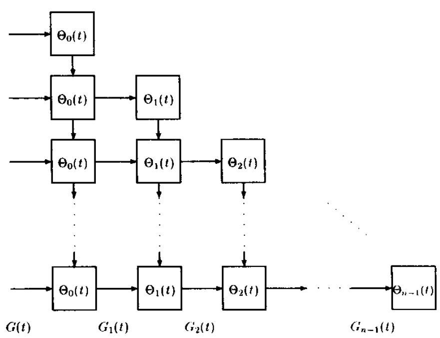


Fig. 8. A triangular array implementation of the generator recursion.

is clear that once the rows of $G(t)$ propagate through the first layer, the array can already receive the rows of $G(t+1)$, etc.

D. An Example: Carathéodory–Fejér Interpolation

We consider here, for the sake of illustration, the time-variant Carathéodory–Fejér example that was described earlier in Section III. This corresponds to choosing $m = 1$, $\alpha_0(t) = 0$, $r_0 = n$, $p = q = 1$, $J(t) = J = (1 \oplus -1)$, and $F(t)$ and $G(t)$ as in (5). The assumption that $\{g_i(t)J(t)g_i^*(t)\}_{t\in\mathbf{Z}}$ be uniformly bounded from below is automatically satisfied here since $f_i(t) = 0$ and hence, $g_i(t)J(t)g_i^*(t) = d_i(t) > b_d > 0$. Using the simplified recursion (24) we can thus write

$$\begin{bmatrix} 0 & 0 \\ G_{i+1}(t) \end{bmatrix} = ZG_i(t-1)\Theta_i(t-1)\begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix} + G_i(t)\Theta_i(t)\begin{bmatrix} 0 & 0 \\ 0 & 1 \end{bmatrix}$$

where $\Theta_i(t)$ is a simple hyperbolic rotation defined by

$$\Theta_i(t) = \frac{1}{\sqrt{1-|\gamma_i(t)|^2}}\begin{bmatrix} 1 & -\gamma_i(t) \\ -\gamma_i^*(t) & 1 \end{bmatrix}, \qquad \gamma_i(t) = \frac{v_{ii}(t)}{u_{ii}(t)}$$

where we denote the entries of the first row of $G_i(t)$ by $g_i(t) = [u_{ii}(t) \;\; v_{ii}(t)]$. The above recursion is the time-variant analogue of the array form associated with the classical Schur algorithm [23], [18].

VII. Applications

The theory developed so far in this paper provides a recursive procedure for the solution of a general time-variant Hermite–Fejér interpolation problem as in (4) (see, e.g., Algorithms 5.3.1 and 6.2.1). It also provides an $O(r(t)n^2)$ procedure for the time-update of the triangular factor $L(t-1)$ to $L(t)$. We now describe several problems in control, model validation, analytic interpolation theory, and adaptive filtering, whose recursive solutions can be obtained as special cases of the algorithm of this paper and correspond to different choices of $F(t)$.

A. Analytic Interpolation Theory

We first show that by specializing Algorithm 6.2.1 to the time-invariant case, we are led to a computationally efficient procedure for the solution of rational analytic interpolation problems, which arise in several applications in circuit theory and control (see, e.g., [12], [21], [24], [25] and the references therein). All we need to do is to drop the time index t. For example, the bounded upper-triangular operator $\mathcal{T}$ now becomes a Toeplitz operator with $r \times r$ entries $T_{|i-j|}$ and can be associated with an $r \times r$ rational matrix function $T(z)$ that is analytic inside the open unit disc ($|z| < 1$), viz.,

$$T(z) = T_0 + T_1 z + T_2 z + \cdots.$$

It is then straightforward to check that definition (3) collapses to a tangential evaluation of the pth derivative of $T(z)$, viz., $\frac{1}{p!}T^{(p)}(f)$ ($|f| < 1$), along the direction of a $1 \times r$ row vector u, viz., $u\frac{1}{p!}T^{(p)}(f)$. Moreover,

$$[u_1 \quad u_2]\mathcal{H}_T^2(f) = [u_1 \quad u_2]\begin{bmatrix} T(f) & \frac{1}{1!}T^{(1)}(f) \\ \mathbf{0} & T(f) \end{bmatrix}.$$

The time-invariant Hermite–Fejér problem can then be stated as follows: consider m stable points $\{\alpha_i\}_{i=0}^{m-1}$ (i.e., inside the open unit disc). We associate with each α_i a positive integer $r_i \geq 1$ and two row vectors $\mathbf{a}_i$ and $\mathbf{b}_i$ partitioned as

$$\mathbf{a}_i = [u_1^{(i)} \quad \cdots \quad u_{r_i}^{(i)}], \qquad \mathbf{b}_i = [v_1^{(i)} \cdots v_{r_i}^{(i)}]$$

where $u_j^{(i)}$ and $v_j^{(i)}$, ($j = 1, \cdots, r_i$), are $1 \times p$ and $1 \times q$ row vectors, respectively.

Problem 7.1.1: Given m stable points $\{\alpha_i\}$ and the associated row vectors $\mathbf{a}_i$ and $\mathbf{b}_i$, describe all rational $p \times q$ Schur functions $S(z)$ (i.e., $S(z)$ analytic and strictly bounded by 1 in $|z| < 1$) that satisfy $\mathbf{b}_i = \mathbf{a}_i \mathcal{H}_S^{r_i}(\alpha_i)$ for $0 \le i \le m-1$.

Following the discussion in Section III, we construct the following (time-invariant) matrices

$$\overline{F}_i = \begin{bmatrix} \alpha_i & & & \\ 1 & \alpha_i & & \\ & \ddots & \ddots & \\ & & 1 & \alpha_i \end{bmatrix}, \qquad U_i = \begin{bmatrix} u_1^{(i)} \\ \vdots \\ u_{r_i}^{(i)} \end{bmatrix},$$

$$V_i = \begin{bmatrix} v_1^{(i)} \\ \vdots \\ v_{r_i}^{(i)} \end{bmatrix}$$

and define $F = \text{diagonal}\{\overline{F}_0, \overline{F}_1, \cdots, \overline{F}_{m-1}\}$

$$G = \begin{bmatrix} U_0 & V_0 \\ \vdots & \vdots \\ U_{m-1} & V_{m-1} \end{bmatrix} \equiv [\mathbf{U} \quad \mathbf{V}], \qquad J = \begin{bmatrix} I_p & \mathbf{0} \\ \mathbf{0} & -I_q \end{bmatrix}.$$

Let $n = \sum_{i=0}^{m-1} r_i$ and $r = (p+q)$, then F and G are $n \times n$ and $n \times r$ matrices, respectively. We also assume $\mathcal{U}\mathcal{U}^* > 0$, where $\mathcal{U} \equiv [\cdots \; F^2\mathbf{U} \; F\mathbf{U} \; \mathbf{U}]$. We further denote the diagonal entries of F by $\{f_i\}_{i=0}^{n-1}$ and consider the following time-invariant displacement equation (compare with (6))

$$R - FRF^* = GJG^*. \tag{27}$$

Theorem 3.1 then collapses to the following.

Theorem 7.1.1: Under the nondegeneracy condition $\mathcal{U}\mathcal{U}^* > 0$, the tangential Hermite–Fejér problem is solvable if, and only if, the solution R of (27) is positive-definite ($R > 0$).

The successive computation of the Schur complements of R in (27) also leads to a recursive update of the generator matrix G. The corresponding proper form (as described in Section VI-B) is as follows: let g_i denote the first row of G_i, which is the generator of the ith Schur complement of R. We now always have $g_i J g_i^* > 0$ since $d_i > 0$ and $d_i = g_i J g_i^*/(1 - |f_i|^2)$. Hence, we can always choose a J-unitary rotation Θ_i that reduces g_i to the form

$$g_i\Theta_i = [\delta_i \quad \mathbf{0}]. \tag{28}$$

In this case, the corresponding generator recursion (24) can be compactly written as follows.

Algorithm 7.1.1: The generators of the successive Schur complements of R in (27) satisfy the recursion

$$\begin{bmatrix} \mathbf{0} \\ G_{i+1} \end{bmatrix} = \Phi_i G_i \Theta_i \begin{bmatrix} 1 & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{bmatrix} + G_i \Theta_i \begin{bmatrix} 0 & \mathbf{0} \\ \mathbf{0} & I_{r-1} \end{bmatrix} \tag{29}$$

where Φ_i is an $(n-i) \times (n-i)$ "Blaschke" matrix, $\Phi_i = (F_i - f_i I_{n-i})(I_{n-i} - f_i^* F_i)^{-1}$. Moreover, the ith column of the triangular factor of R is given by $l_i = (I_{n-i} - f_i^* F_i)^{-1} G_i \Theta_i [\delta_i \; \mathbf{0}]^{\mathbf{T}}$. The generator recursion (29) has the following simple array interpretation: multiply G_i by Θ_i and keep the last $r-1$ columns; multiply the first column of $G_i\Theta_i$ by Φ_i; these two steps result in G_{i+1}. Each step of (29) also leads to a first-order J-lossless section, which we shall denote by $T_i(z)$. This can be easily seen by writing down (24) in the present case

$$\begin{bmatrix} l_i & \mathbf{0} \\ & G_{i+1} \end{bmatrix} = [F_i l_i \quad G_i] \begin{bmatrix} f_i^* & \frac{\delta_i}{d_i}[1 \quad \mathbf{0}] \\ \Theta_i \begin{bmatrix} \delta_i \\ \mathbf{0} \end{bmatrix} & \Theta_i \begin{bmatrix} -f_i & \mathbf{0} \\ \mathbf{0} & I \end{bmatrix} \end{bmatrix}. \tag{30}$$

This defines a first-order section $T_i(z)$

$$\begin{aligned} T_i(z) &= \Theta_i \begin{bmatrix} -f_i & \mathbf{0} \\ \mathbf{0} & I \end{bmatrix} + \Theta_i \begin{bmatrix} \delta_i \\ \mathbf{0} \end{bmatrix} (z^{-1} - f_i^*)^{-1} \frac{\delta_i}{d_i} [1 \quad \mathbf{0}] \\ &= \Theta_i \begin{bmatrix} \frac{z - f_i}{1 - z f_i^*} & \mathbf{0} \\ \mathbf{0} & I \end{bmatrix}. \end{aligned} \tag{31}$$

Observe that the blocking property of each such section is very evident since

$$g_i T_i(f_i) = g_i \Theta_i \begin{bmatrix} 0 & \mathbf{0} \\ \mathbf{0} & I \end{bmatrix} = [\delta_i \quad \mathbf{0}] \begin{bmatrix} 0 & \mathbf{0} \\ \mathbf{0} & I \end{bmatrix} = \mathbf{0}.$$

As in the time-variant case, the local blocking properties reflect on the entire cascade $T(z) = T_0(z) \cdots T_{n-1}(z)$, and lead to the following parameterization result: partition $T(z)$ accordingly with J

$$T(z) = \begin{bmatrix} T_{11}(z) & T_{12}(z) \\ T_{21}(z) & T_{22}(z) \end{bmatrix}.$$

Theorem 7.1.2: All solutions $S(z)$ of the tangential Hermite–Fejér problem are given through a linear fractional transformation of a Schur matrix function $K(z)$

$$S(z) = -[T_{11}(z)K(z) + T_{12}(z)] \cdot [T_{21}(z)K(z) + T_{22}(z)]^{-1}.$$

We are thus led to the following $O(rn^2)$ recursive solution of the time-invariant Hermite–Fejér algorithm (see [13], [15], [16] for more details along the lines of this paper).

Algorithm 7.1.2: The Hermite–Fejér Problem 7.1.1 can be recursively solved as follows: construct F, G, and J from the interpolation data as described in the beginning of this section. Start with $G_0 = G$, $F_0 = F$, and repeat for $i = 0, 1, \cdots, n-1$.

- At step i we have F_i and G_i.
- Use a convenient J-unitary rotation Θ_i that reduces g_i to the proper form (28). This determines δ_i and the first-order section $T_i(z)$ as in (31).

The cascade of all n sections parameterizes all solutions $S(z)$ as described in the above theorem.

B. Time-Domain Model Validation

We now consider another application that arises in the problem of model validation. In robust control, it is often the case that uncertainty models are used to cope with the lack of an exact mathematical model for a physical plant (see, e.g., [27], [28]). In this context, an uncertainty model is usually described in terms of a known nominal model, represented by an upper-triangular operator $\mathcal{M}_0$, a modeling uncertainty, represented by an upper-triangular operator $\mathcal{S}$, and the form by which $\mathcal{M}_0$ and $\mathcal{S}$ are combined. A simple example is that of additive dynamic uncertainty [27], viz., an uncertainty model of the form $(\mathcal{M}_0 + \mathcal{W}\mathcal{S})$, where $\mathcal{S}$ is constrained to be a (strictly) contractive operator and $\mathcal{W}$ is a known weighting function. The associated model validation problem can then be stated as follows: assume we apply an input sequence to the physical system, say $\{u_0, u_1, \cdots, u_{n-1}\}$, where $\{u_i\}$ is a $1 \times p$ row input vector at time i, and measure the associated noisy output sequence, say $\{v_0, v_1, \cdots, v_{n-1}\}$, where $\{v_i\}$ is a $1 \times q$ row output vector at time i. The question is then to verify whether there exists a model for the plant of the form $(\mathcal{M}_0 + \mathcal{W}\mathcal{S})$ that maps the given input sequence to the measured output sequence

$$[v_0 \quad v_1 \quad \cdots \quad v_{n-1}] = [u_0 \quad u_1 \quad \cdots \quad u_{n-1}](\mathcal{M}_0 + \mathcal{W}\mathcal{S}).$$

This clearly reduces to checking whether there exists a strict contraction $\mathcal{S}$ mapping the signals:

$$[v_0 \quad v_1 \quad \cdots \quad v_{n-1}] - [u_0 \quad u_1 \quad \cdots \quad u_{n-1}]\mathcal{M}_0 \quad \text{and } [u_0 \quad u_1 \quad \cdots \quad u_{n-1}]\mathcal{W}.$$

More involved examples are considered in [27], [28]. As above, a major step in the solution is to check whether there exists an upper-triangular strictly contractive operator that maps two sets of signals. But this is a special case of the following general time-variant Carathéodory–Fejér problem: given data points $\{u_i(t), v_i(t)\}_{t\in\mathbf{Z}}$, $i = 0, 1, \cdots, n-1$, it is required to find conditions for the existence of an upper-triangular strict contraction $\mathcal{S} = [S_{ij}]$ such that

$$[u_0(t-n+1) \quad \cdots \quad u_{n-1}(t)] \cdot \begin{bmatrix} S_{t-n+1,t-n+1} & \cdots & & S_{t-n+1,t} \\ & \ddots & & \vdots \\ & & S_{t-1,\,t-1} & S_{t-1,\,t} \\ & & & S_{tt} \end{bmatrix} = [v_0(t-n+1) \quad \cdots \quad v_{n-1}(t)].$$

Hence, the recursive algorithm developed in this paper (Algorithm 5.3.1 and especially its simplified form in Algorithm 6.2.1) provides a recursive solution for the model validation problem.

To put this into our framework, we construct a displacement equation as in (6) with

$$\mathbf{U}(t) = \begin{bmatrix} u_0(t) \\ \vdots \\ u_{n-1}(t) \end{bmatrix}, \qquad \mathbf{V}(t) = \begin{bmatrix} v_0(t) \\ \vdots \\ v_{n-1}(t) \end{bmatrix},$$

$$F(t) = Z, \qquad G(t) = [\mathbf{U}(t) \quad \mathbf{V}(t)].$$

Theorem 7.2.1: The tangential Carathéodory–Fejér problem has solutions if, and only if, there exists a real number $\epsilon > 0$, independent of t, such that $R(t) = \mathcal{U}(t)\mathcal{U}^*(t) - \mathcal{V}(t)\mathcal{V}^*(t) > \epsilon I$ for all $t \in \mathbf{Z}$, where

$$\mathcal{U}(t) \equiv \begin{bmatrix} & & & u_0(t) \\ & & u_0(t-1) & u_1(t) \\ & \cdot & & \vdots \\ u_0(t-n+1) & \cdots & u_{n-2}(t-1) & u_{n-1}(t) \end{bmatrix}$$

and a similar expression for $\mathcal{V}(t)$ with $v_i(t)$ instead of $u_i(t)$.

In applications where data is available over a finite period of time, say $|t| < N$, the condition in the statement of the theorem is equivalent to the positivity of $R(t)$, viz., $R(t) > 0$ for $|t| < N$. The recursion of Algorithm 6.2.1 then provides an efficient procedure for testing this condition. It computes, for each t, the diagonal entries $\{d_0(t), \cdots, d_{n-1}(t)\}$ and the positivity condition $R(t) > 0$ is satisfied if, and only if, $d_i(t) > 0$ for $i = 0, \cdots, n-1$.

We finally remark that the condition of strict contractivity on $\mathcal{S}$ can be relaxed as discussed in [17], [29].

C. Strong Parrott's Problem

We mentioned earlier that a rather general extension of Theorem 3.1 is possible [17], [29], where the entries of $R(t)$ are regarded as linear operators acting on appropriate Hilbert spaces, and the positive-definiteness condition is relaxed to positive-semidefiniteness ($R(t) \geq 0$). An immediate application that fits into this framework is the so-called strong Parrott problem, which arises in the study of the spectral properties of the key "four-block" operator in control (see, e.g., [30]–[33]): given matrices B_{ij}, $1 \leq j \leq i \leq n$, $S = [S_1 \; S_2 \; \cdots \; S_n]$ and $T = [T_1 \; T_2 \; \cdots \; T_n]$, it is required to find conditions for the existence of a contraction $\mathcal{T}$ of the form (? denotes unspecified entries)

$$\mathcal{T} = \begin{bmatrix} B_{11} & & & \\ B_{21} & B_{22} & & ? \\ \vdots & & \ddots & \\ B_{n1} & B_{n2} & \cdots & B_{nn} \end{bmatrix}$$

such that $S\mathcal{T} = T$. To put this problem into our framework, we define

$$\mathbf{U}(t) = \begin{bmatrix} \mathbf{0} \\ I \\ \mathbf{0} \\ \vdots \\ \mathbf{0} \end{bmatrix}, \qquad 1 \leq t \leq n-1,$$

$$\mathbf{U}(t) = \begin{bmatrix} S_{n+t} \\ I \\ \mathbf{0} \\ \vdots \\ \mathbf{0} \end{bmatrix}, \qquad -n+1 \leq t \leq 0,$$

$$\mathbf{V}(0) = \begin{bmatrix} T_1 \\ B_{n1} \\ B_{n-1,1} \\ \vdots \\ B_{11} \end{bmatrix}, \qquad \mathbf{V}(1) = \begin{bmatrix} T_2 \\ \mathbf{0} \\ B_{n2} \\ \vdots \\ B_{22} \end{bmatrix},$$

$$\mathbf{V}(2) = \begin{bmatrix} T_3 \\ \mathbf{0} \\ \mathbf{0} \\ B_{n3} \\ \vdots \\ B_{33} \end{bmatrix}, \cdots, \mathbf{V}(n-1) = \begin{bmatrix} T_n \\ \mathbf{0} \\ \vdots \\ \mathbf{0} \\ B_{nn} \end{bmatrix}$$

$$F(t) = \begin{bmatrix} I & & & & & \\ \mathbf{0} & \mathbf{0} & & & & \\ & I & \mathbf{0} & & & \\ & & I & \mathbf{0} & & \\ & & & \ddots & \ddots & \\ & & & & I & \mathbf{0} \end{bmatrix},$$

$$G(t) = [\mathbf{U}(t) \quad \mathbf{V}(t)], \quad \text{for } -n+1 \le t \le n-1$$

and all the elements equal to zero for the other time indices. We then consider the associated time-variant displacement equation as in (6). A generalization of Theorem 3.1 (as in [17], [29]) then shows the following.

Lemma 7.3.1: The above strong Parrott problem has solutions if, and only if, the solution $R(t)$ of (6) satisfies $R(t) \geq 0$ for all $t \in \mathbb{Z}$.

It is easy to check that the above condition is equivalent to that stated in [30], [33].

D. The QR Algorithm in Adaptive Filtering and Control

In adaptive filtering and control, one is often faced with the task of solving a linear system of (so-called normal) equations of the form $\Phi(t)w(t) = b(t)$, where $\Phi(t)$ is an $n \times n$ positive-definite Hermitian matrix usually referred to as the autocorrelation matrix, $b(t)$ is an $n \times 1$ column vector known as the cross-correlation vector, and $w(t)$ is an $n \times 1$ so-called weight vector. The reader is referred to [26] for more on the derivation and motivation of the normal equations in the context of the recursive least-squares problem.

It often happens (e.g., in an exponential windowing scheme) that the quantities $\Phi(t)$ and $b(t)$ satisfy time recursions of the form $\Phi(t) - \lambda\Phi(t-1) = u^*(t)u(t)$, $b(t) - \lambda b(t-1) = x(t)u^*(t)$, where λ is a positive scalar $(0 < \lambda \leq 1)$, $u(t)$ is a row vector, and $x(t)$ is a scalar. In the adaptive applications, one is often faced with the problem of solving $\Phi(t)w(t) = b(t)$, for successive time instants t, $t+1$, $t+2, \cdots$, and one expects a computationally efficient procedure for going through the calculations, $w(t) \to w(t+1) \to w(t+2) \to \cdots$ since the corresponding quantities, $(\Phi(t), b(t))$, $(\Phi(t+1), b(t+1))$, $(\Phi(t+2), b(t+2)), \cdots$ are closely related. For example, $\Phi(t+1)$ and $\lambda\Phi(t)$ differ only by a rank-one matrix $u^*(t+1)u(t+1)$.

We will now show how to exploit the existing time-variant displacement (or low-rank) structure to derive a fully parallelizable solution of the normal equations and one that avoids the drawbacks of approaches based on back-substitution techniques. The discussion that follows exploits the fact that the generator recursion in Algorithm 6.2.1 also propagates the triangular factor of the time-variant matrix $R(t)$.

The main point is to start by expressing $w(t)$ as a Schur complement in a suitable block matrix and then to properly exploit the structure of this matrix. Consider the following square $(2n \times 2n)$ extended matrix

$$R(t) = \begin{bmatrix} \Phi(t) & b(t) & \mathbf{0} \\ I_n & \mathbf{0} & \mathbf{0} \end{bmatrix}$$

and note that the Schur complement of $\Phi(t)$ in $R(t)$ is precisely $[-w(t) \;\; \mathbf{0}]$, which completely identifies the desired weight vector $w(t)$. This already suggests the following route: if we can show how to efficiently go from the Schur complement at time $t-1$ to the Schur complement at time t, then we obtain an efficient procedure for going from $w(t-1)$ to $w(t)$. But this is precisely what is provided by Algorithm 6.2.1. It follows from the displacement structures of $\Phi(t)$ and $b(t)$ that $R(t)$ is also structured, viz.,

$$R(t) - \begin{bmatrix} \lambda I_n & \mathbf{0} \\ \mathbf{0} & I_n \end{bmatrix} R(t-1) = \begin{bmatrix} u^*(t) \\ \mathbf{0} \end{bmatrix} [u(t) \quad x(t) \quad \mathbf{0}]$$

which is a special case of a non-Hermitian time-variant displacement equation of the form $R(t) - F(t)R(t-1)A^*(t) = G(t)J(t)B^*(t)$, with $A(t) = I_{2n}$, $J(t) = 1$, $\Delta = 1$

$$F(t) = \begin{bmatrix} \lambda I_n & \mathbf{0} \\ \mathbf{0} & I_n \end{bmatrix}, \qquad G(t) = \begin{bmatrix} u^*(t) \\ \mathbf{0} \end{bmatrix}, \quad \text{and}$$

$$B(t) \begin{bmatrix} u^*(t) \\ x(t) \\ \mathbf{0} \end{bmatrix}.$$

Although $R(t)$ has a non-Hermitian structure, we are only interested in its first n Schur complementation steps (that is, in its first n triangular factors). But the leading $n \times n$ submatrix of $R(t)$ is Hermitian (equal to $\Phi(t)$) and hence, Algorithm 6.2.1 is still applicable. Following this argument, we can show (see [16, 18] for more details) that the simplified generator Algorithm 6.2.1 collapses to the following.

Algorithm 7.4.1: The solution of the normal equations that arise in the recursive-least squares problem can be recursively updated as follows: form the prearray of numbers on the left-hand side of the equation below; choose any unitary matrix $\Theta(t)$ that produces the zero block in the postarray; the other entries in the postarray can then be shown to be the quantities needed for the next time-instant plus the entries denoted by $\bar{e}(t)$ and $\overline{\Delta w}(t)$, which are used in updating the solution $w(t)$

$$\begin{bmatrix} \sqrt{\lambda}\Phi^{1/2}(t-1) & u^*(t) \\ \sqrt{\lambda}b^*(t-1)\Phi^{-*/2}(t-1) & x^*(t) \\ \frac{1}{\sqrt{\lambda}}\Phi^{-*/2}(t-1) & \mathbf{0} \end{bmatrix} \Theta(t) = \begin{bmatrix} \Phi^{1/2}(t) & \mathbf{0} \\ b^*(t)\Phi^{-*/2}(t) & \bar{e}(t) \\ \Phi^{-*/2}(t) & \overline{\Delta w}(t) \end{bmatrix}$$

and $w(t)$ is time-updated via $w(t) = w(t-1) - \overline{\Delta w}(t)\bar{e}^*(t)$. ∎

The above algorithm provides a parallel procedure for the extraction of the weight vector $w(t)$ and avoids the drawbacks of a back-substitution step (see, e.g., [18] for more details and for the corresponding systolic architecture). It is also worth noting here that the normal equations correspond to a particular diagonal choice of $F(t)$, namely, $F(t) = (\lambda I \oplus I)$. The results of this paper however, are applicable to more general matrices $F(t)$ as discussed in [16], [18].

VIII. Concluding Remarks

We posed and solved a general time-variant Hermite–Fejér interpolation problem by an approach that handles the time-variant case in much the same way as it handled the time-invariant counterpart [14]–[16]. The derivation is based on a simple matrix calculation known as the Schur reduction procedure and leads to a recursive construction of a time-variant cascade or transmission-line structure that makes evident the interpolation properties. State-space descriptions for each section and for the entire cascade were also derived in terms of the original interpolation data and a time-variant structured matrix $R(t)$. The cascade is constructed recursively by implicitly considering the triangular factorization of $R(t)$. Unlike other approaches, we do not require explicit knowledge of the matrices $R(t)$ or $R^{-1}(t)$. The whole recursive procedure works only with the matrices $F(t)$ and $G(t)$ that are constructed directly from the interpolation data. The overall computational complexity is then $O(r(t)n^2)$ operations per time step, where $r(t)$ is a so-called displacement rank (the number of columns of the matrix $G(t)$). Under a supplementary condition, we further obtained a substantial simplification of the cascade structure and presented a triangular array of lattice sections that solves the interpolation problem. We then considered applications in adaptive filtering, robust control, model validation, and analytic interpolation. Other applications to the solution of matrix completion problems are also possible (see e.g., [13], [17], [29]).

Appendix A
Addendum to Theorem 3.1

We check that the interpolation conditions (4) and the relation $\mathcal{V}(t) = \mathcal{U}(t)\hat{\mathcal{S}}(t)$ are indeed equivalent. This follows easily by expanding $\mathcal{V}(t)$ and $\mathcal{U}(t)$. For example, the first entries of $\mathcal{V}(t)$ and $\mathcal{U}(t)$ corresponding to the first Jordan block $\overline{F}_0(t)$ are of the form

$$\mathcal{V}(t) = \begin{bmatrix} \cdots & f_0(t)v_1^{(0)}(t-1) & v_1^{(0)}(t) \\ \cdots & v_1^{(0)}(t-1) + f_0(t)v_2^{(0)}(t-1) & v_2^{(0)}(t) \\ & v_2^{(0)}(t-1) + f_0(t)v_3^{(0)}(t-1) & v_3^{(0)}(t) \\ & \vdots & \vdots \end{bmatrix},$$

$$\mathcal{U}(t) = \begin{bmatrix} \cdots & f_0(t)u_1^{(0)}(t-1) & u_1^{(0)}(t) \\ \cdots & u_1^{(0)}(t-1) + f_0(t)u_2^{(0)}(t-1) & u_2^{(0)}(t) \\ & u_2^{(0)}(t-1) + f_0(t)u_3^{(0)}(t-1) & u_3^{(0)}(t) \\ & \vdots & \vdots \end{bmatrix}.$$

We now use (3) to obtain the desired result.

Appendix B
Proof of Lemma 5.3.1

We verify here that $\mathcal{T}_{12}^{(i)}\mathcal{T}_{22}^{-(i)}$ is an upper-triangular strictly contractive operator. We first remark that because of the $\mathcal{J}$-losslessness property of Lemma 5.1.2 we conclude that

$$\mathcal{T}_{22}^{(i)}\mathcal{T}_{22}^{*(i)} \geq I \quad \text{and} \quad \mathcal{T}_{22}^{*(i)}\mathcal{T}_{22}^{(i)} \geq I \tag{B.1}$$

which implies that $\mathcal{T}_{22}^{(i)}$ is an invertible operator and that $\mathcal{T}_{22}^{-(i)}$ is contractive. That is, $\|\mathcal{T}_{22}^{-(i)}\| \leq 1$. We already know that $\mathcal{T}_{22}^{(i)}$ is upper-triangular (by construction). We need to verify that its inverse is also upper-triangular. Let $X^{-(n)}$ denote the following leading subsection of $\mathcal{T}_{22}^{(i)}$

$$X^{-(n)} = \begin{bmatrix} \ddots & & & \vdots \\ & T_{22,i}^{n-2,n-2} & T_{22,i}^{n-2,n-1} & T_{22,i}^{n-2,n} \\ \mathbf{0} & & T_{22,i}^{n-1,n-1} & T_{22,i}^{n-1,n} \\ & & & T_{22,i}^{nn} \end{bmatrix}.$$

It then follows from (B.1) that

$$X^{-*(n)}X^{-(n)} \geq I. \tag{B.2}$$

Similarly, we define $\mathcal{Y}^{(n)}$ and $\mathcal{J}_n$ to denote the corresponding leading subsections of $\mathcal{T}_i$ and $\mathcal{J}$, respectively. It follows from the embedding relation (12) that we can write

$$\mathcal{J}_n - \mathcal{Y}^{(n)}\mathcal{J}_n\mathcal{Y}^{*(n)} = \begin{bmatrix} \vdots \\ g^*(n-1)f^*(n) \\ g^*(n) \end{bmatrix} d^{-1}(n) \cdot [\cdots \quad f(n)g(n-1) \quad g(n)] \geq 0. \tag{B.3}$$

But $X^{-(n)}$ is the (2, 2) block entry in $\mathcal{Y}^{(n)}$, i.e.,

$$\mathcal{Y}^{(n)} = \begin{bmatrix} ? & ? \\ ? & X^{-(n)} \end{bmatrix}.$$

Hence, we conclude from (B.3) that $X^{-(n)}$ also satisfies

$$X^{-(n)}X^{-*(n)} \geq I. \tag{B.4}$$

Combining (B.2) and (B.4) we conclude that $X^{-(n)}$ is invertible for every n and that the inverses $X^{(n)}$ are uniformly bounded by one, i.e., $\|X^{(n)}\| \leq 1$ for all n. Therefore, $X^{(n)}$ is a sequence of bounded operators. Define the following operators (acting on the same space as $\mathcal{T}_{22}^{(i)}$)

$$\tilde{X}^{(n)} = \begin{bmatrix} X^{(n)} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{bmatrix}.$$

Then $\tilde{X}^{(n)}$ and $\tilde{X}^{(n+1)}$ satisfy the following nested property (they differ by just one block column)

$$\tilde{X}^{(n+1)} = \begin{bmatrix} \tilde{X}^{(n)} & ? \\ \mathbf{0} & ? \end{bmatrix}. \tag{B.5}$$

We now verify that the sequence $\{\tilde{X}^{(n)}\}_{n\in\mathbf{Z}}$ is strongly convergent. That is, there exists an operator X: $\mathcal{X} \to \mathcal{Y}$ (where $\mathcal{X}$ and $\mathcal{Y}$ are appropriate Hilbert spaces), such that $\lim_{n\to\infty} \|a\tilde{X}^{(n)} - aX\| = 0$ for all $a \in \mathcal{X}$. By the strong operator convergence theorem [34], we only need to check that the sequence $\{a\tilde{X}^{(n)}\}$ is Cauchy in $\mathcal{Y}$ for every a in a total subset of $\mathcal{X}$. That is, for every $\epsilon > 0$, there exists an N_ϵ such that $\|a\tilde{X}^{(n)} - a\tilde{X}^{(m)}\| < \epsilon$ for every n, $m > N_\epsilon$. To verify this, we consider the following sequence of vectors $\{a_j\}$, $a_j \in \mathcal{X}$ (which is dense in $\mathcal{X}$), $a_j = [\cdots\ 0\ 0\ ?\ ?\ \cdots]$, where a_j has an increasing number of zeros (as j increases). For every j, we set $N_\epsilon = j$ and consider all n and m greater than N_ϵ. Clearly (using the nesting property (B.5) and choosing, without loss of generality, $m > n$), $a_j(\tilde{X}^{(n)} - \tilde{X}^{(m)}) = 0$. That is, $\|a_j\tilde{X}^{(n)} - a_j\tilde{X}^{(m)}\| = 0$ for every $N_\epsilon = j$ and $n, m > j$, which shows that $\{a_j\tilde{X}^{(n)}\}$ is indeed a Cauchy sequence. Therefore, $\tilde{X}^{(n)}$ converges strongly to X. It also follows that $\tilde{X}^{(n)}$ converges weakly to X. Hence, $\lim_{n\to\infty} \langle \tilde{X}^{(n)}e_j, e_i\rangle = \langle Xe_j, e_i\rangle$, where $\langle\,,\,\rangle$ denotes inner product and $\{e_i\}$ are the basis vectors. Choosing $i < j$ we readily conclude that X is also an upper-triangular operator.

In summary, we proved that the inverses $\tilde{X}^{(n)}$ converge strongly to a bounded upper-triangular operator X. We now verify that the operators $\mathcal{T}_{22}^{-(i)}$ and X coincide over a dense subset of their domain of definition. This follows immediately by considering again the sequence of inputs a_j and observing that $a_j\mathcal{T}_{22}^{(i)}X = a_j$ for every j. Therefore, we proved that $\mathcal{T}_{22}^{(i)}$ is invertible and that its inverse is an upper-triangular strict contraction. Finally, it follows from Lemma 5.1.2 that $\mathcal{T}_{12}^{(i)}\mathcal{T}_{22}^{-(i)}$ is also an upper-triangular strict contraction.

Appendix C
Values of $h_i(t)$ and $k_i(t)$

For each time instant t, we consider a transfer matrix $\Theta_i(z, t)$ defined by the expression

$$\Theta_i(z, t) = J(t)k_i^*(t)J(t) + J(t)g_i^*(t)[z^{-1} - f_i^*(t)]^{-1}h_i^*(t)J(t).$$

It follows from the embedding relation (12) that

$$\Theta_i(z, t)J(t)\Theta_i^*(z, t) = J(t) \text{ on the circle } zz^* = \frac{d_i(t-1)}{d_i(t)}. \tag{C.6}$$

Let $\tau_i(t)$ be an arbitrary point on this circle. We first show how to choose a pair $(\hat{h}_i(t), \hat{k}_i(t))$ such that the corresponding transfer matrix, $\hat{\Theta}_i(z, t)$, would satisfy $\hat{\Theta}_i(\tau_i(t), t) = I_{r(t)}$, or equivalently, $\hat{k}_i(t) + \hat{h}_i(t)[\tau_i^{-*}(t) - f_i(t)]^{-1}g_i(t) = I_{r(t)}$. It follows from (12) that $\hat{h}_i(t)d_i(t-1)f_i^*(t) + \hat{k}_i(t)J(t)g_i^*(t) = \mathbf{0}$. Therefore, we can solve for $\hat{h}_i(t)$ and $\hat{k}_i(t)$

$$\hat{h}_i(t) = \frac{1 - \tau_i^*(t)f_i(t)}{\tau_i^*(t)d_i(t) - d_i(t-1)f_i^*(t)}J(t)g_i^*(t),$$
$$\hat{k}_i(t) = I_{r(t)} - \frac{\tau_i^*(t)J(t)g_i^*(t)g_i(t)}{\tau_i^*(t)d_i(t) - d_i(t-1)f_i^*(t)}.$$

The claim is that all other possible choices of $h_i(t)$ and $k_i(t)$ are related to $\hat{h}_i(t)$ and $\hat{k}_i(t)$ by $h_i(t) = \Theta_i^{-1}(t)\hat{h}_i(t)$ and $k_i(t) = \Theta_i^{-1}(t)\hat{k}_i(t)$, for an arbitrary $J(t)$-unitary matrix $\Theta_i(t)$. To check this, let $\Theta_i(z, t)$ be the transfer matrix of any other possible choice $(h_i(t), k_i(t))$. Clearly $\Theta_i(\tau_i(t), t)J(t)\Theta_i^*(\tau_i(t), t) = J(t)$, since $\tau_i(t)$ is a point on the appropriate circle. If we define $\hat{\Theta}_i(z, t) \equiv \Theta_i(z, t)\Theta_i^{-1}(\tau_i(t), t)$, then $\hat{\Theta}_i(\tau_i(t), t) = I_{r(t)}$. Using the fact that this condition is satisfied by $(\hat{h}_i(t), \hat{k}_i(t))$ above, we readily conclude that $h_i(t) = \Theta_i^{-1}(\tau_i(t), t)\hat{h}_i(t)$ and $k_i(t) = \Theta_i^{-1}(\tau_i(t), t)\hat{k}_i(t)$.

Appendix D
Proof of the Uniform Boundedness of $h_i(t)$ and $k_i(t)$

We use the boundedness of $\{d_i(t)\}_{t\in\mathbf{Z}}$ (as stated in Lemma 4.1.1, (15)) and the freedom in choosing $\tau_i(t)$ and $\Theta_i(t)$, to guarantee the uniform boundedness of $h_i(t)$ and $k_i(t)$. For this purpose, we fix $\Theta_i(t) = I_{r(t)}$ and consider the quantity

$$\tau_i^*(t)d_i(t) - d_i(t-1)f_i^*(t) \tag{D.7}$$

which appears in the denominator of the expressions for $h_i(t)$ and $k_i(t)$. Observe that

$$\begin{aligned}|\tau_i^*(t)d_i(t) - d_i(t-1)f_i^*(t)| &= \left|\tau_i^*(t) - \frac{d_i(t-1)}{d_i(t)}f_i^*(t)\right|\frac{1}{d_i(t)}\\ &> \left|\tau_i^*(t) - \frac{d_i(t-1)}{d_i(t)}f_i^*(t)\right|\frac{1}{c_d}\end{aligned}$$

and recall that $\tau_i(t)$ is an arbitrary point in the complex plane to be chosen on the circle of radius $\sqrt{\frac{d_i(t-1)}{d_i(t)}}$. So assume we choose a point $\tau_i(t)$ that is in the opposite direction of $f_i(t)$ (we can also choose $\tau_i(t)$ to be in the direction orthogonal to $f_i(t)$). Then we clearly have

$$\begin{aligned}\left|\tau_i^*(t) - \frac{d_i(t-1)}{d_i(t)}f_i^*(t)\right|\frac{1}{c_d} &> \frac{|\tau_i^*(t)|}{c_d}\\ &= \frac{1}{c_d}\sqrt{\frac{d_i(t-1)}{d_i(t)}} > \sqrt{\frac{b_d}{c_d^3}}.\end{aligned}$$

This shows that (D.7) can always be uniformly bounded from below. It then follows from the uniform boundedness of $\{f_i(t), g_i(t)\}_{t\in\mathbf{Z}}$ that $\{h_i(t), k_i(t)\}_{t\in\mathbf{Z}}$ can always be chosen to be uniformly bounded sequences.

References

[1] A. Feintuch and B. A. Francis, "Distance formulas for operator algebras arising in optimal control problems," *Operator Theory: Advances and Applications*, vol. 29, pp. 151–170, 1988.

[2] A. Feintuch and B. A. Francis, "Uniformly optimal control of linear time-varying systems," *Syst. Contr. Lett.*, vol. 5, pp. 67–71, 1984.

[3] P. P. Khargonekar and M. A. Rotea, "Coprime factorization for linear time-varying systems," in *Proc. IEEE Amer. Contr. Conf.*, Atlanta, GA, 1988, pp. 848–851.

[4] M. A. Rotea and P. P. Khargonekar, "Stabilizability of linear time-varying and uncertain linear systems," *IEEE Trans. Automat. Contr.*, vol. 33, pp. 884–887, 1988.

[5] R. Ravi, K. M. Nagpal, and P. P. Khargonekar, "H_∞ control of linear time-varying systems: A state-space approach," *SIAM J. Contr. Optim.*, vol. 29, pp. 1394–1423, 1991.

[6] W. B. Arveson, "Interpolation problems in nest algebras," *J. Functional Analysis*, vol. 20, pp. 208–233, 1975.

[7] J. A. Ball and I. Gohberg, "A commutant lifting theorem for triangular matrices with diverse applications," *Integral Equations and Operator Theory*, vol. 8, pp. 205–267, 1985.

[8] D. Alpay, P. Dewilde, and H. Dym, "Lossless inverse scattering and reproducing kernels for upper triangular operators," *Operator Theory: Advances and Applications*, vol. 47, pp. 61–135, 1990.

[9] P. Dewilde, "A course on the algebraic Schur and Nevanlinna-Pick interpolation problems," in Ed. F. Deprettere and A. J. van der Veen, Eds., *Algorithms and Parallel VLSI Architectures*, vol. A. Amsterdam: Elsevier, 1991, pp. 13–69.

[10] P. Dewilde and H. Dym, "Interpolation for upper triangular operators," *Operator Theory: Advances and Applications*, vol. 56, pp. 153–260, 1992.

[11] J. A. Ball, I. Gohberg, and M. A. Kaashoek, "Nevanlinna–Pick interpolation for time-varying input–output maps: The discrete case," *Operator Theory: Advances and Applications*, vol. 56, pp. 1–51, 1992.

[12] J. A. Ball, I. Gohberg, and L. Rodman, *Interpolation of Rational Matrix Functions.* Basel: Birkhäuser, 1990.

[13] A. H. Sayed, T. Constantinescu, and T. Kailath, "Square-root algorithms for structured matrices, interpolation, and completion problems," *IMA Volumes in Mathematics and Its Applications*, to appear.

[14] A. H. Sayed, T. Kailath, H. Lev-Ari, and T. Constantinescu, "Recursive solutions of rational interpolation problems via fast matrix factorization," *Integral Equations and Operator Theory,* to appear.

[15] A. H. Sayed and T. Kailath, "Recursive solutions to rational interpolation problems," in *Proc. IEEE Intern. Symp. Circuits & Syst.*, vol. 5, San Diego, CA, May 1992, pp. 2376–2379.

[16] A. H. Sayed, "Displacement structure in signal processing and mathematics." PhD dissertation, Stanford Univ., Stanford, CA, Aug. 1992.

[17] T. Constantinescu, A. H. Sayed, and T. Kailath, "Displacement structure and completion problems," *SIAM J. Matrix Anal. and Appl.*, to appear.

[18] A. H. Sayed, H. Lev-Ari, and T. Kailath, "Time-variant displacement structure and triangular arrays," *IEEE Trans. Sig. Proc.*, to appear.

[19] I. Gohberg and M. A. Kaashoek, "Time-varying linear systems with boundary conditions and integral operators, I. The transfer operator and its applications," *Integral Equations and Operator Theory*, vol. 7, pp. 325–391, 1984.

[20] T. Kailath, *Linear Systems.* Englewood Cliffs, NJ: Prentice-Hall, 1980.

[21] H. Dym, *J-Contractive Matrix Functions, Reproducing Kernel Hilbert Spaces and Interpolation*, vol. 71. Providence, RI: American Mathematical Society, 1989.

[22] H. Lev-Ari and T. Kailath, "State-space approach to factorization of lossless transfer functions and structured matrices," *Linear Algebra and Its Applications*, vol. 162–164, pp. 273–295, Feb. 1992.

[23] T. Kailath ,"A theorem of I. Schur and its impact on modern signal processing," *Operator Theory: Advances and Applications*, vol. 18, pp. 9–30, 1986.

[24] H. Kimura, "Directional interpolation approach to H^{∞}-optimization and robust stabilization," *IEEE Trans. Automat. Contr.*, vol. AC-32, no. 12, pp. 1085–1093, Dec. 1987.

[25] D. J. N. Limebeer and B. D. O. Anderson, "An interpolation theory approach to H^{∞}-controller degree bounds," *Linear Algebra and Its Applications*, vol. 98, pp. 347–386, 1988.

[26] S. Haykin, *Adaptive Filter Theory*, 2nd ed. Englewood Cliffs, NJ: Prentice-Hall, 1991.

[27] K. Poolla, P. Khargonekar, A. Tikku, J. Krause, and K. Nagpal, "A time-domain approach to model validation," in *Proc. American Contr. Conf.*, vol. 1, Chicago, IL, June 1992, pp. 313–317.

[28] R. S. Smith and J. C. Doyle, "Model validation: A connection between robust control and identification," *IEEE Trans. Automat. Contr.*, vol. 37, no. 7, pp. 942–952, July 1992.

[29] T. Constantinescu, A. H. Sayed, and T. Kailath, "Structured matrices and moment problems," in *Proc. of the Workshop on Challenges of a Generalized System Theory*, Amsterdam, Netherlands, June 1992, pp. 25–43.

[30] C. Foias and A. Tannenbaum, "A strong Parrott theorem," in *Proc. of the AMS*, vol. 106, 1989, pp. 777–784.

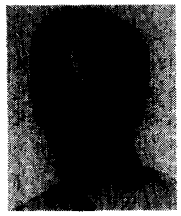
[31] C. Foias and A. Tannenbaum, "On the four block problem I," *Operator Theory: Advances and Applications*, vol. 32, pp. 93–122, 1986.

[32] ——, "On the four block problem II: The singular system," *Integral Equations and Operator Theory*, vol. 11, pp. 726–767, 1988.

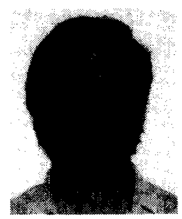
[33] M. Bakonyi and H. J. Woerdeman, "On the strong Parrott completion problem," in *Proc. Amer. Math. Soc.*, vol. 117, 1993, pp. 429–433.

[34] A. H. Sayed, T. Constantinescu, and T. Kailath, "Lattice structures for time-variant interpolation problems," in *Proc. IEEE Conf. Decision & Contr.*, vol. 1, Tucson, Arizona, Dec. 1992, pp. 116–121.

[35] F. Riesz and B. Sz. Nagy, *Functional Analysis.* New York: Ungar, 1955.



Ali H. Sayed (S'90–M'92) was born in São Paulo, Brazil, in 1963. In 1981 he graduated in first place in the National Lebanese Baccalaureat. In 1987 he received the degree of Engineer in electrical engineering from the University of São Paulo, Brazil, and was the first-place graduate of the School of Engineering. In 1989 he received, with distinction, the M.S. degree in electrical engineering from the University of São Paulo, Brazil, and in 1992 he received the Ph.D. in electrical engineering from Stanford University, Stanford, CA. From September 1992 to August 1993, he was a Research Associate with the Information Systems Laboratory at Stanford University. He is currently an Assistant Professor in the Department of Electrical and Computer Engineering at the University of California, Santa Barbara. Dr. Sayed's current research interests are in the areas of statistical, digital and adaptive signal processing, robust filtering and control, and structured computations in systems and mathematics.



Tiberiu Constantinescu was born in Bucharest, Romania, on June 13, 1955. He received the M.S. and Ph.D. degrees in Mathematics from the University of Bucharest in 1980 and 1989, respectively. From 1980 to 1989 he was a Researcher at INCREST, Bucharest, after which he joined the Institute of Mathematics of the Romanian Academy, Bucharest. From 1991 to 1993 he was a visiting scholar at the Information Systems Laboratory, Stanford University. He is currently with the University of Texas at Dallas. Dr. Constantinescu's research interests are in applications of operator theory and interpolation problems.

Thomas Kailath (S'57–M'62–F'70) was educated in Poona, India, and at the Massachusetts Institute of Technology (S.M., 1959; Sc.D., 1961). From October 1961 to December 1962, he worked at the Jet Propulsion Laboratories, Pasadena, CA, where he also taught part time at the California Institute of Technology. He then came to Stanford University, where he served as Director of the Information Systems Laboratory from 1971 through 1980, as Associate Department Chairman from 1981 to 1987, and currently holds the Hitachi America Professorship in Engineering. He has held short-term appointments at several institutions around the world. Dr. Kailath has worked in a number of areas including information theory, communications, computation, control, signal processing, VLSI design, statistics, linear algebra and operator theory; his recent interests include applications of signal processing, computation and control to problems in semiconductor manufacturing and wireless communications. He is the author of *Linear Systems* (Prentice Hall, 1980) and *Lectures on Wiener and Kalman Filtering* (Springer-Verlag, 1981). He has held Guggenheim, Churchill and Royal Society fellowships, among others, and received awards from the IEEE Information Theory Society and the American Control Council, in addition to the Technical Achievement and Society Awards of the IEEE Signal Processing Society. He served as President of the IEEE Information Theory Society in 1975, and has been awarded honorary doctorates by Linköping University, Sweden, and by Strathclyde University, Scotland. He is a Fellow of the IEEE and of the Institute of Mathematical Statistics and is a member of the National Academy of Engineering.