INCOMING AND OUTGOING ALGORITHMS FOR MATRIX PADÉ APPROXIMATION AND MULTIVARIABLE AR FILTERING

A. BULTHEEL

Katholieke Universiteit Leuven, Applied Mathematics and Programming Division, Celestijnenlaan 200 A, B-3030 Heverlee, Belgium

Abstract

The approximation of a transfer function from a related Taylor or Laurent series (spectral information) is treated by means of recursive computation of elements in a matrix Padé table. A bordering technique to solve a block Toeplitz system is used to derive recursive procedures that are incoming and outgoing versions of the algorithms for the computation of a row or a pair of rows in the Padé table. The theory of Laurent-Padé approximation is then used to show that these are generalizations of Levinson-Wiggins-Robinson and Schur algorithms, well known in linear prediction theory. Known results are brought together in a systematic way. Numerical examples are given.

Permission to publish this abstract separately is granted.

1. Introduction

The technique of Padé approximation and continued fraction expansion has been used in linear filtering theory in a number of situations [1-5]. However in those situations one usually gets involved in problems of filter-stability. The attractive simplicity of recursive computations have inspired people to take a Padé approximant as an initial guess and then force it to be stable. For autoregressive filtering there exist the Szegö-Levinson and Schur algorithms that share the recursive computational techniques of Padé approximation and at least theoretically deliver stable approximations and in practical applications it is simply controlled by the magnitude of the reflection coefficients.

In this paper we show that these Levinson and Schur algorithms are special cases of incoming and outgoing recursions for the computation of rows in a Laurent-Padé table. Par. 2 gives the algorithms for Padé-row computations of the incoming type like the Levinson algorithm, while par. 3 gives the corresponding outgoing recursions that generalize the Schur algorithm. Par. 4 introduces the Laurent-Padé problem and it is shown there that the algorithms of previous sections still work for this case. In the last section it is shown that the Levinson and Schur algorithms appear as special cases. Numerical examples illustrate how the algorithms work.

Most of the individual results of the paper are found, scattered in the literature. We attempt to bring them together in a systematic way and clarify the relation between two important techniques in signal processing.

2. Row Padé recursions : a review

In this section we quickly review some basic definitions and a recursive algorithm for the computation of rows in the Padé table. This will be of incoming type and generalize the Levinson-Wiggins-Robinson recursion.

Let be given a formal power series

$$F(z) = f_0 + f_1 z + f_2 z^2 + \ldots$$

with coefficients f_k in the ring of $N \times N$ complex matrices. Define $A^{(m/n)}(z) = (P^{(m/n)}(z))(Q^{(m/n)}(z))^{-1}$ with $P^{(m/n)}(z)$ and $Q^{(m/n)}(z)$ $N \times N$ matrix polynomials of degree m and n respectively and without a non constant common right divisor. $A^{(m/n)}(z)$ is called a right Padé approximant (RPA) of $F(z)$ if their formal power series expansions agree through as many terms as possible (at least m+n+1). So we must have

$$R^{(m/n)}(z) \triangleq F(z)Q^{(m/n)}(z) - P^{(m/n)}(z) = r_0^{(m/n)} z^{m+n+1} + r_1^{(m/n)} z^{m+n+2} + \ldots .$$

$R^{(m/n)}(z)$ is called a right residual.

Let

$$P^{(m/n)}(z) \triangleq p_0^{(m/n)} + p_1^{(m/n)} z + \ldots + p_m^{(m/n)} z^m$$

$$Q^{(m/n)}(z) \triangleq q_0^{(m/n)} + q_1^{(m/n)} z + \ldots + q_n^{(m/n)} z^n ,$$

then the following system of equations is obtained :

$$\sum_{j=0}^{n} f_{i-j}\, q_j^{(m/n)} = \begin{cases} p_i^{(m/n)} & \text{for } i = 0,\ldots,m \\ 0 & \text{for } i = m+1,\ldots,m+n \\ r_{i-(m+n+1)}^{(m/n)} & \text{for } i = m+n+1,\ldots \end{cases} \qquad (2.1)$$

Define $f_k = 0$ for $k < 0$ and let $T(k,i,j)$ denote a block Toeplitz matrix, based on the coefficients $\{f_k\}$ as follows :

$$T(k,i,j) \triangleq \begin{bmatrix} f_k & f_{k-1} & \cdots & f_{k-i} \\ f_{k+1} & & & \\ \vdots & & & \\ f_{k+j} & \cdots & & f_{k+j-i} \end{bmatrix} , \quad k,i,j \geq 0 . \qquad (2.2)$$

Furthermore set

$$\mathbb{Q}^{(m/n)} = [q_0^{(m/n)} \ \dots \ q_n^{(m/n)}]^t$$

$$\mathbb{P}^{(m/n)} = [p_0^{(m/n)} \ \dots \ p_m^{(m/n)}]^t$$

$$\mathbb{R}^{(m/n)} = [r_0^{(m/n)} \quad r_1^{(m/n)} \ \dots]^t$$

where ^{t} denotes block transpose from now on.

System (2.1) now becomes :

$$\begin{bmatrix} T(0,n,m) \\ T(m+1,n,n-1) \\ T(m+n+1,n,\infty) \end{bmatrix} \mathbb{Q}^{(m/n)} = \begin{bmatrix} \mathbb{P}^{(m/n)} \\ 0 \\ \mathbb{R}^{(m/n)} \end{bmatrix} . \tag{2.3}$$

We shall suppose that $F(z)$ is normal, i.e. that all $T(k,i,j)$ are nonsingular. In that case $F(z)^{-1}$ and all RPA's exist and all of $p_0^{(m/n)}$, $p_m^{(m/n)}$, $q_0^{(m/n)}$, $q_n^{(m/n)}$ and $r_0^{(m/n)}$ are nonsingular. This also allows to remove the ambiguity in the definition of the RPA. We shall multiply $P^{(m/n)}(z)$ and $Q^{(m/n)}(z)$ with a constant matrix, such that we have one of the following normalizations : $q_n^{(m/n)} = I$ or $q_0^{(m/n)} = I$. In the latter case, $\mathbb{Q}^{(m/n)}$ is determined by

$$T(m+1,n-1,n-1)[q_1^{(m/n)} \ \dots \ q_n^{(m/n)}]^t = -[f_{m-n+1} \ f_{m-n+2} \ \dots \ f_m]^t . \tag{2.4}$$

In any event, by adding a block row and a block column to (2.4), the following equivalent forms are obtained :

$$T(m+1,n,n)\mathbb{Q}^{(m/n)} = [0 \ \dots \ 0 \ r_0^{(m/n)}]^t \tag{2.5}$$

$$T(m,n,n)\mathbb{Q}^{(m/n)} = [p_n^{(m/n)} \ 0 \ \dots \ 0]^t . \tag{2.6}$$

Instead of the RPA $A^{(m/n)}(z)$ with R-numerator and R-denominator, we can also introduce a <u>left Padé approximant</u> (LPA) $^{(m/n)}A(z) = (^{(m/n)}Q(z))^{-1}(^{(m/n)}P(z))$ with L-numerator $^{(m/n)}P(z)$ and L-denominator $^{(m/n)}Q(z)$. The definition is identical, except that the L-residual is now defined by a left multiplication :

$$ {}^{(m/n)}R(z) \triangleq {}^{(m/n)}Q(z)F(z) - {}^{(m/n)}P(z) = {}^{(m/n)}r_0\, z^{m+n+1} + \ldots \ldots $$

Using a similar notation as above, (2.5) and (2.6) transform into

$$ {}^{(m/n)}\mathbb{Q}^t\, T(m+1,n,n)^t = [0 \ldots 0\ {}^{(m/n)}r_0] \qquad (2.7) $$

$$ {}^{(m/n)}\mathbb{Q}^t\, T(m,n,n)^t = [{}^{(m/n)}p_n\ 0 \ldots 0] \ldots \qquad (2.8) $$

Remark that systems (2.5) and (2.7) e.g. are essentially different since $(AB)^t \neq B^t A^t$ for general block matrices. Only in the scalar case they do coincide.

The block Toeplitz structure of T(k,i,j) implies that the normality condition for L- and R-approximations are the same and it takes also little effort to show that the RPA and LPA are the same [6] so we speak of a Padé approximant (PA) and denote it by $[m/n](z) = A^{(m/n)}(z) = {}^{(m/n)}A(z)$. However, remark that there exist distinct R and L numerators and denominators.

There is a systematic way to construct recursive solutions for (2.5) - (2.8), using a bordering technique [7]. We give a quick indication of how to use this technique in appendix 1 and give here the main results for the computation of Padé-rows which will generalize the multivariable Levinson-Schur algorithm.

We use the following abbreviations for fixed m and m+1, selecting rows m and m+1 in the Padé table :

$$ T_n \triangleq T(m+1,n,n),\ \underline{T}_n \triangleq T(m,n,n),\ \tilde{T}_n \triangleq T(m+2,n,n)\ , $$

$$ \mathbb{Q}_n \triangleq \mathbb{Q}^{(m/n)} \quad \text{with monic normalization : } q_n^{(m/n)} = I $$

$$ \bar{\mathbb{Q}}_n \triangleq \mathbb{Q}^{(m+1/n)} \quad \text{with comonic normalization : } q_0^{(m+1/n)} = I $$

$$ \mathbb{P}_n \triangleq \mathbb{P}^{(m/n)}\ ,\ p_n = p_m^{(m/n)}\ ,\ \bar{\mathbb{P}}_n \triangleq \mathbb{P}^{(m+1/n)}\ ,\ \vec{p}_n \triangleq p_{m+1}^{(m+1/n)}\ , $$

$$\mathbb{R}_n \triangleq \mathbb{R}^{(m/n)} \quad , \quad r_n \triangleq r_0^{(m/n)} \quad , \quad r'_n \triangleq r_1^{(m/n)} \quad ,$$

$$\bar{\mathbb{R}}_n \triangleq \mathbb{R}^{(m+1/n)} \quad , \quad \bar{r}_n \triangleq r_0^{(m+1/n)} \quad , \quad \bar{r}'_n \triangleq r_1^{(m+1/n)} \quad .$$

Note that the upper bar normalization is different from the no bar normalization for reasons which will soon become apparent. A similar notation is used for the L-quantities, shifting the index from right to left. Note also that ${}_nT = T_n^t$.

A right continued fraction (RCF) is denoted by

$$a_0 + \left\lfloor \frac{a_1}{b_1} \right. + \left\lfloor \frac{a_2}{b_2} \right. + \ldots = a_0 + \sum_{k=1}^{\infty} \left\lfloor \frac{a_k}{b_k} \right.$$

and is defined to be the (ordered) set of convergents : $\{C_n = P_n \, Q_n^{-1}\}_0^\infty$ where

$$P_{-1} = I, \; P_0 = a_0, \; P_{i+1} = P_{i-1} \, a_{i+1} + P_i \, b_{i+1} \qquad \text{for } i \geqslant 0$$

$$Q_{-1} = 0, \; Q_0 = I, \; Q_{i+1} = Q_{i-1} \, a_{i+1} + Q_i \, b_{i+1} \qquad \text{for } i \geqslant 0.$$

A left continued fraction (LCF) is denoted as

$$\ldots + \left\lfloor \frac{{}^{2}a}{{}^{2}b} \right. + \left\lfloor \frac{{}^{1}a}{{}^{1}b} \right. + \; {}^{0}a = \sum_{k=1}^{\infty} \left\lfloor \frac{{}^{k}a}{{}^{k}b} \right. + \; {}^{0}a \; .$$

It means the set of convergents $\{{}_nC = {}_nQ^{-1} \; {}_nP\}_0^\infty$ with

$${}_{-1}P = I, \; {}_0P = {}_0a, \; {}_{i+1}P = {}_{i+1}a \; {}_{i-1}P + {}_{i+1}b \; {}_iP \qquad \text{for } i \geqslant 0 \; ,$$

$${}_{-1}Q = 0, \; {}_0Q = I, \; {}_{i+1}Q = {}_{i+1}a \; {}_{i-1}Q + {}_{i+1}b \; {}_iQ \qquad \text{for } i \geqslant 0 \; .$$

From now on we also use a nomenclature that has to be understood in block sense wherever appropriate. E.g. a row means block row, triangular means block triangular etc. .

The following recursion gives $\mathbb{Q}_{n+1}$ from $\mathbb{Q}_n$ and $\mathbb{Q}_{n-1}$ which can be derived by using a bordering technique (see appendix 1)

$$\mathbb{Q}_{n+1} = \begin{bmatrix} 0 \\ \\ \mathbb{Q}_n \end{bmatrix} - \left(\begin{bmatrix} 0 \\ \mathbb{Q}_{n-1} \\ 0 \end{bmatrix} - \begin{bmatrix} \mathbb{Q}_n \\ \\ 0 \end{bmatrix} r_n^{-1} r_{n-1} \right) p_{n-1}^{-1} p_n \qquad (2.9)$$

$$n = 0,1,2,\ldots$$

with initial conditions

$\mathbb{Q}_{-1} = [\emptyset]$, the empty vector ,

$p_{-1} = I$, $r_{-1} = -I$,

$\mathbb{Q}_0 = I$, $p_0 = f_m$ and $r_0 = f_{m+1}$.

We call this scheme an incoming recursion because in every step, more information comes in and the amount of work increases as the dimension of $\mathbb{Q}$ increases. The coefficients p_n and r_n, needed in the recursion are given by

$$\begin{bmatrix} p_n \\ \\ r_n \end{bmatrix} = \begin{bmatrix} f_m & \cdots & f_{m-n} \\ & & \\ f_{m+n+1} & \cdots & f_{m+1} \end{bmatrix} \mathbb{Q}_n ,$$

where f_{m-n} and f_{m+n+1} are the new pieces of information that enter the recursion.

It follows immediately from (2.3) that we have the following triangular factorizations

$$T_n \begin{bmatrix} I & & \\ & \begin{bmatrix} \mathbb{Q}_i \end{bmatrix} & \star \\ 0 & & I \end{bmatrix} = \begin{bmatrix} r_0 & & \\ & \begin{bmatrix} \mathbb{Z}_i \end{bmatrix} & 0 \\ \star & & r_n \end{bmatrix} \qquad (2.10)$$

and

$$\tilde{T}_n \begin{bmatrix} \star & & 0 \\ & \begin{bmatrix} \mathbb{Q}_i \end{bmatrix} & \\ I & & I \end{bmatrix} = \begin{bmatrix} p_n & & \\ & \begin{bmatrix} \mathbb{S}_i \end{bmatrix} & \star \\ 0 & & p_0 \end{bmatrix} \qquad (2.11)$$

where the $\mathbb{Q}_i$ are as above and $\mathbb{Z}_i$ and $\mathbb{S}_i$ are defined by the relations (2.10) - (2.11). $\mathbb{Z}_i$ contains R-residual coefficients and $\mathbb{S}_i$ R-numerator coefficients. These are the outgoing vectors. Their length decreases as the length of the $\mathbb{Q}_i$ vectors decreases. $\mathbb{Z}_0$ and $\mathbb{S}_0$ contain all the information at the beginning and they are contracted at the moment that some information is extracted.

By replacing m by m+1 and upper-baring all quantities, a similar recursion is available for $\bar{\mathbb{Q}}_n$, giving the factorizations

$$\bar{T}_n \begin{bmatrix} I & & & \\ & \begin{bmatrix} & & \\ & \bar{\mathbb{Q}}_i & \\ & & \end{bmatrix} & & I \\ 0 & & \ddots & \\ & & & \star \end{bmatrix} = \begin{bmatrix} \bar{r}_0 & & & 0 \\ & \ddots & & \\ \star & & \begin{bmatrix} & & \\ & \mathbb{Z}_i & \\ & & \end{bmatrix} & \\ & & & \bar{r}_n \end{bmatrix} \tag{2.12}$$

and

$$T_n \begin{bmatrix} I & & & \\ & \ddots & & 0 \\ & & \begin{bmatrix} & & \\ & \bar{\mathbb{Q}}_i & \\ & & \end{bmatrix} & \\ \star & & & I \end{bmatrix} = \begin{bmatrix} \bar{p}_n & & & \star \\ & \ddots & \begin{bmatrix} & & \\ & \bar{\mathbb{S}}_i & \\ & & \end{bmatrix} & \\ 0 & & & \bar{p}_0 \end{bmatrix} \tag{2.13}$$

Remark that the change in normalization assures here a lower triangular in (2.13) with I's on the diagonal. This was the reason for the different normalization of the upper bar quantities.

We could reformulate everything in its L-version. Formally, this can be obtained by shifting the indices from right to left and taking the block transpose, using the rules of an ordinary transpose.

Instead of looking for the three term recursion with the elements based on one row, we could look for recursions that split this up in two parts, in a sawtooth fashion as follows :

$$\mathbb{Q}_n \;\&\; \bar{\mathbb{Q}}_n \overset{(1)}{\Rightarrow} \mathbb{Q}_{n+1} \;;\; \mathbb{Q}_{n+1} \;\&\; \bar{\mathbb{Q}}_n \overset{(2)}{\Rightarrow} \bar{\mathbb{Q}}_{n+1} \;;\; \mathbb{Q}_{n+1} \;\&\; \bar{\mathbb{Q}}_{n+1} \overset{(1)}{\Rightarrow} \mathbb{Q}_{n+2} \quad \text{etc..}$$

In step (1) we seek to express $\mathbb{Q}_{n+1}$ as a function of $\mathbb{Q}_n$ and $\bar{\mathbb{Q}}_n$, and in step (2) we seek $\bar{\mathbb{Q}}_{n+1}$ from $\mathbb{Q}_{n+1}$ and $\bar{\mathbb{Q}}_n$. The following recursive scheme is derived using the bordering technique (appendix 1)

$$\mathbb{Q}_0 = I \;,\; \bar{\mathbb{Q}}_0 = I \;,\; r_0 = f_{m+1}$$

$$(1)\quad \mathbb{Q}_{n+1} = \begin{bmatrix} 0 \\ \mathbb{Q}_n \end{bmatrix} - \begin{bmatrix} \bar{\mathbb{Q}}_n \\ 0 \end{bmatrix} \bar{p}_n^{-1}\, p_n \;;\; r_{n+1} = r_n - \bar{r}_n\, \bar{p}_n^{-1}\, p_n \qquad (2.14)$$

$$(2)\quad \bar{\mathbb{Q}}_{n+1} = (\mathbb{Q}_{n+1} - \begin{bmatrix} \bar{\mathbb{Q}}_n \\ 0 \end{bmatrix} \bar{r}_n^{-1}\, r_{n+1})\nu_n \;;\; \bar{p}_{n+1} = -\bar{p}_n\, \bar{r}_n^{-1}\, r_{n+1}\, \nu_n$$

$$\nu_n = (q_0^{(m/n+1)} - \bar{r}_n^{-1}\, r_{n+1})^{-1}$$

$$p_n = [f_m \;\dots\; f_{m-n}]\mathbb{Q}_n \;;\; \bar{r}_n = [f_{m+n+2} \;\dots\; f_{m+n}]\bar{\mathbb{Q}}_n \;.$$

ν_n was introduced to preserve the proper normalization on $\mathbb{Q}_n$. Now the new pieces of information are introduced one by one. In step (1) we introduce f_{m-n} via p_n and in step (2) we inject f_{m+n+2} via $\bar{r}_n$.

We could also introduce these coefficients in the other order. This results in a dual sawtooth algorithm which goes like :

$$\mathbb{Q}_n \;\&\; \bar{\mathbb{Q}}_n \overset{(1)}{\Rightarrow} \bar{\mathbb{Q}}_{n+1} \;;\; \bar{\mathbb{Q}}_{n+1} \;\&\; \mathbb{Q}_n \overset{(2)}{\Rightarrow} \mathbb{Q}_{n+1} \;;\; \mathbb{Q}_{n+1} \;\&\; \bar{\mathbb{Q}}_{n+1} \overset{(1)}{\Rightarrow} \mathbb{Q}_{n+2} \;\cdots$$

The resulting recursion is (appendix 1) :

$$Q_0 = I \; ; \quad \bar{Q}_n = I \; , \; \bar{p}_0 = f_{m+1} \; ,$$

$$(1) \quad \bar{Q}_{n+1} = \begin{bmatrix} \bar{Q}_n \\ 0 \end{bmatrix} - \begin{bmatrix} 0 \\ Q_n \end{bmatrix} r_n^{-1} \bar{r}_n \; ; \; \bar{p}_{n+1} = \bar{p}_n - p_n r_n^{-1} \bar{r}_n \tag{2.15}$$

$$(2) \quad Q_{n+1} = (\bar{Q}_{n+1} - \begin{bmatrix} 0 \\ Q_n \end{bmatrix} p_n^{-1} \bar{p}_{n+1})\lambda_n ; \; r_{n+1} = -r_n p_n^{-1} \bar{p}_{n+1}$$

with $\lambda_n = (\bar{q}_{n+1}^{(m+1/n+1)} - p_n^{-1} \bar{p}_{n+1})^{-1}$, such that the canonic normalization for $\bar{Q}_{n+1}$ is obtained, and p_n and $\bar{r}_n$ are as in (2.14).

A mixed scheme of (2.14) and (2.15) gives the analog of the Levinson scheme for Padé approximations :

$$Q_{n+1} = \begin{bmatrix} 0 \\ Q_n \end{bmatrix} + \begin{bmatrix} \bar{Q}_n \\ 0 \end{bmatrix} \alpha_n \; , \; \alpha_n = -\bar{p}_n^{-1} p_n , \; r_{n+1} = r_n(I-\beta_i \alpha_i) \; ,$$

$$\bar{Q}_{n+1} = \begin{bmatrix} \bar{Q}_n \\ 0 \end{bmatrix} + \begin{bmatrix} 0 \\ Q_n \end{bmatrix} \beta_n \; , \; \beta_n = -r_n^{-1} \bar{r}_n , \; \bar{p}_{n+1} = \bar{p}_n(I-\alpha_i \beta_i) \; , \tag{2.16}$$

and r_0, $\bar{p}_0$, p_i and $\bar{r}_i$ as before.

Writing out polynomials, one has

$$[Q_{n+1}(z) \; \bar{Q}_{n+1}(z)] = [Q_n(z) \; \bar{Q}_n(z)] \begin{bmatrix} zI & 0 \\ 0 & I \end{bmatrix} \begin{bmatrix} I & \beta_n \\ \alpha_n & I \end{bmatrix}$$

$$= [I \quad I] \; \theta_0 \; \ldots \; \theta_n = [I \quad I] \; \Theta_n \tag{2.17}$$

where

$$\theta_i(z) = \begin{bmatrix} zI & O \\ O & I \end{bmatrix} \begin{bmatrix} I & \beta_i \\ \alpha_i & I \end{bmatrix}.$$

Again the L-denominators for row m and m+1 can be obtained by similar algorithms. The same rule of thumb as given before yields the result. As an example, the L-version of (2.17) is :

$$\begin{bmatrix} {}_{n+1}Q(z) \\ {}_{n+1}\bar{Q}(z) \end{bmatrix} = {}_n\theta \begin{bmatrix} {}_nQ(z) \\ {}_n\bar{Q}(z) \end{bmatrix} = {}_n\theta \cdots {}_0\theta \begin{bmatrix} I \\ I \end{bmatrix} = {}_n\Theta \begin{bmatrix} I \\ I \end{bmatrix} \qquad (2.18)$$

with $${}_i\theta(z) = \begin{bmatrix} I & {}_i\alpha \\ {}_i\beta & I \end{bmatrix} \begin{bmatrix} zI & O \\ O & I \end{bmatrix}.$$

The R- and L-algorithms can be obtained in an efficient way to give the two triangular factors of T_n^{-1} and ${}_nT^{-1} = (T_n^t)^{-1}$. For more details, consult [7].

3. Outgoing or Schur-type recursions

In the previous section we gave algorithms of incoming type. They all build up denominator polynomials of increasing degree. They are also characterized by the fact that they require the computation of an inner product of a Toeplitz row and a denominator vector to obtain diagonal elements in the right hand side triangular factors of (2.10) - (2.13). Because of the linearity of the latter relations, we are able to transfer the recursions from previous section for the columns of the left hand side to recursions for the columns on the right hand side. We thus obtain recursions for the outgoing vectors $\mathbb{Z}_i$, $\mathbb{S}_i$ and/or $\bar{\mathbb{Z}}_i$ and $\bar{\mathbb{S}}_i$.

At the start $\mathbb{Z}_0$ and $\mathbb{S}_0$ contain all the information stored in T_n. In fact, if $n = \infty$, we have the whole formal power series in it and we shall construct a RCF expansion for it. In order to obtain practical schemes, we fix for a moment n, the maximal order of approximation and define <u>"cutting" operators</u> to construct vectors of decreasing length :

$$\lrcorner \triangleq \left[\begin{array}{ccc|c} I & & 0 & \\ & \ddots & & 0 \\ 0 & & I & \end{array}\right] \quad \text{and} \quad \urcorner \triangleq \left[\begin{array}{c|ccc} & I & & 0 \\ 0 & & \ddots & \\ & 0 & & I \end{array}\right] .$$

(2.9) is then translated into the following RCF constructing algorithm which is of outgoing type [7]

$$\begin{array}{l|l}
\mathbb{Z}^t_{-1} = [I \ \underbrace{0 \ \dots \ 0}_{n+1}] & \mathbb{S}^t_{-1} = [\underbrace{0 \ \dots \ 0}_{n+1} \ I] \\
\mathbb{Z}^t_0 = [f_{m+1} \ \dots \ f_{m+n+1}] & \mathbb{S}^t_0 = [f_{m-n} \ \dots \ f_m] \\
\mathbb{Z}_{i+1} = \lrcorner\mathbb{Z}_i + \urcorner(\lrcorner\mathbb{Z}_{i-1} + \mathbb{Z}_i u_i) v_i & \mathbb{S}_{i+1} = \urcorner(\lrcorner\mathbb{S}_{i-1} + \mathbb{S}_i u_i) v_i + \lrcorner\mathbb{S}_i \\
u_i = -r_i^{-1} r_{i-1} & v_i = -p_{i-1}^{-1} p_i \\
r_i = [I \ 0 \ \dots \ 0]\mathbb{Z}_i & p_i = [0 \ \dots \ 0 \ I]\mathbb{S}_i
\end{array} \qquad (3.1)$$

Remark that there is no inner product left in this scheme.

The underlying RCF that may be constructed from the u_i, v_i-coefficients is :

$$F(z) = P^{(m/0)}(z) + z^m \sum_{i=0}^{\infty} \frac{z \, v_i}{|z + u_i v_i} \quad , \qquad (3.2)$$

with as j-th convergent the PA : $P^{(m/j)}(z)(Q^{(m/j)}(z))^{-1}$.

(3.2) is a contracted form of the RCF

$$P^{(m/0)}(z) + z^m \cfrac{z\, f_{m+1}\rfloor}{\lceil I} + \cfrac{z\rfloor}{\lceil \mu_1} + \cfrac{\nu_i\rfloor}{\lceil I} + \cfrac{z\rfloor}{\lceil \mu_2} + \cfrac{\nu_2\rfloor}{\lceil I} + \ldots \qquad (3.3)$$

with $\mu_{i+1} = -\bar{p}_i^{-1} p_i$, $\nu_{i+1} = -\bar{r}_i^{-1} r_{i+1}$ and p_i, $\bar{p}_i$, r_i and $\bar{r}_i$ are found from the recursive scheme that is the outgoing analog of (2.14) viz.

$$\mathbb{Z}_0 = [f_{m+1} \ \ldots \ f_{m+n+1}]^t \qquad \mathbb{S}_0 = [f_{m-n} \ \ldots \ f_m]^t$$

$$\bar{\mathbb{Z}}_0 = [f_{m+2} \ \ldots \ f_{m+n+2}]^t \qquad \bar{\mathbb{S}}_0 = [f_{m-n+1} \ \ldots \ f_{m+1}]^t$$

for $i = 0,1,2,\ldots$ (3.4)

$$\mathbb{Z}_{i+1} = \mathbb{Z}_i + \bar{\mathbb{Z}}_i \, \mu_{i+1} \qquad \mathbb{S}_{i+1} = \lrcorner(\mathbb{S}_i + \bar{\mathbb{S}}_i \, \mu_{i+1})$$

$$\bar{\mathbb{Z}}_{i+1} = \urcorner(\mathbb{Z}_{i+1} + \bar{\mathbb{Z}}_i \, \nu_{i+1}) \qquad \bar{\mathbb{S}}_{i+1} = \mathbb{S}_{i+1} + \urcorner\bar{\mathbb{S}}_i \, \nu_{i+1}$$

$$[r_i \ \bar{r}_i] = [I \ \ 0 \ \ldots \ 0][Z_i \ \bar{Z}_i] \qquad [p_i \ \bar{p}_i] = [0 \ \ldots \ 0 \ \ I][\mathbb{S}_i \ \bar{\mathbb{S}}_i]$$

We consider next the outgoing recursion for (2.16) which is the generalization of the Schur algorithm :

$$r_0 = f_{m+1} = \bar{p}_0$$

$$\bar{r}_0 = f_{m+2} \qquad\qquad p_0 = f_m$$

$$\mathbb{Z}_0^t = [f_{m+1} \ \ldots \ f_{m+n+1}] \qquad\qquad \mathbb{S}_0^t = [f_{m-n} \ \ldots \ f_m]$$

$$\bar{\mathbb{Z}}_0^t = [f_{m+2} \ \ldots \ f_{m+n+2}] \qquad\qquad \bar{\mathbb{S}}_0^t = [f_{m-n+1} \ \ldots \ f_{m+1}]$$

for $i = 0,1,2,\ldots,n-1$

$$\beta_i = -r_i^{-1}\,\bar{r}_i \qquad\qquad \alpha_i = -\bar{p}_i^{-1}\,p_i \tag{3.5}$$

$$\bar{\mathbb{Z}}_{i+1} = \neg(\bar{\mathbb{Z}}_i + \mathbb{Z}_i\,\beta_i) \qquad\qquad \mathbb{S}_{i+1} = \lrcorner(\mathbb{S}_i + \bar{\mathbb{S}}_i\,\alpha_i)$$

$$\mathbb{Z}_{i+1} = \lrcorner(\mathbb{Z}_i + \bar{\mathbb{Z}}_i\,\alpha_i) \qquad\qquad \bar{\mathbb{S}}_{i+1} = \neg(\bar{\mathbb{S}}_i + \mathbb{S}_i\,\beta_i)$$

$$r_{i+1} = [I \ 0 \ \ldots \ 0]\mathbb{Z}_{i+1} = [0 \ \ldots \ 0 \ I]\mathbb{S}_{i+1} = \bar{p}_{i+1}$$

$$\bar{r}_{i+1} = [I \ 0 \ \ldots \ 0]\,\bar{\mathbb{Z}}_{i+1} \qquad\qquad p_{i+1} = [0 \ \ldots \ 0 \ I]\mathbb{S}_{i+1}$$

The following compact form can be used for the residual/numerator recursions of (3.5). Take $n = \infty$ and define for $i = 0,1,2,\ldots$ the residuals

$$R_i(z) = z^{m+i+1}\,[I \ zI \ z^2I \ \ldots]\mathbb{Z}_i \ ,$$

$$\bar{R}_i(z) = z^{m+i+2}\,[I \ zI \ z^2I \ \ldots]\bar{\mathbb{Z}}_i$$

and the numerators

$$P^{(m/i)}(z) = S_i(z) = [\ldots\ z^{-1}I\ \ I\ \ zI\ \ \ldots\ \ z^{m}I]\mathbb{S}_i \quad \text{and}$$

$$P^{(m+1/i)}(z) = \tilde{S}_i(z) = [\ldots\ z^{-1}I\ \ I\ \ zI\ \ \ldots\ \ z^{m+1}I]\bar{\mathbb{S}}_i\ , \text{ then}$$

$$\begin{bmatrix} R_{n+1}(z) & \bar{R}_{n+1}(z) \\ S_{n+1}(z) & \bar{S}_{n+1}(z) \end{bmatrix} = \begin{bmatrix} R_n(z) & \bar{R}_n(z) \\ S_n(z) & \bar{S}_n(z) \end{bmatrix} \theta_n = \begin{bmatrix} R_0(z) & \bar{R}_0(z) \\ S_0(z) & \bar{S}_0(z) \end{bmatrix} \Theta_n \qquad (3.6)$$

with θ_i and Θ_n as defined in (2.17). Remark that the terms in z^{-k} do not appear in $S_i(z)$ and $\tilde{S}_i(z)$ since they have zero coefficients in $\mathbb{S}_i$ and $\bar{\mathbb{S}}_i$. These quantities are related to the RCF (3.2) in the following way :

$$F(z) = \sum_{0}^{m} f_k z^k + z^m \sum_{i=1}^{n} \left| \frac{z\ v_i}{z+u_i v_i} \right| - \left| \frac{R_{n+1}(k)}{R_n(z)} \right|$$

and

$$P^{(m/n)}(z)\,(Q^{(m/n)}(z))^{-1} = (S_n(z))(Q_n(z))^{-1}$$

$$= \sum_{0}^{m} f_k z^k + z^m \sum_{i=1}^{n} \left| \frac{z\ v_i}{z+u_i v_i} \right| \quad .$$

Example 1

We specialize to the scalar case for the sake of simplicity and compactness and illustrate the computational schemes with a simple example.

Take $F(z) = (z+1.1)^{-1}(z-3.3)^{-1}(z^2+1.4^2)^{-1} = \sum_{k=0}^{\infty} f_k z^k$ with f_k given in table 1 :

k	f_k
0	-0.14055209
1	0.08518308
2	-0.01863545
3	0.03476062
4	-0.06278763
5	0.04762904
6	-0.02749619
7	0.02978531

table 1

The right hand side factors of (2.10 - 2.13) contain the vectors $\mathbb{Z}_i$, $\bar{\mathbb{Z}}_i$, $\mathbb{S}_i$ and $\bar{\mathbb{S}}_i$, produced by algorithm (3.5).

For m = 0 and n = 4, they are given in table 2 :

i	0	1	2	3	4
$\mathbb{Z}_i$	0.08518304				
	-0.01863545	0.05443460			
	0.03476062	0.03871958	0.1851584		
	-0.06278764	-0.06883898	-0.1963802	-0.06388731	
	0.04762904	0.01580028	0.07555752	0.03781958	0.0

table 2.a

i	0	1	2	3	4
$\bar{\mathbb{Z}}_i$	-0.01863545				
	0.03476062	0.03068376			
	-0.06278764	-0.05518308	-0.07700858		
	0.04762904	0.03389303	0.07269627	-0.008979495	
	-0.02749619	-0.01707642	-0.02598274	0.005442118	0.0

table 2.b

i	0	1	2	3	4
$\mathbb{S}_i$	1.0	0.0	0.0	0.0	0.0
		-0.4545455	0.0	0.0	0.0
			-0.5988024	0.0	0.0
				-0.2319109	0.0
					-0.1405521

table 2.c

i	0	1	2	3	4
$\bar{\mathbb{S}}_i$	0.0	-0.1405521	0.0	0.0	0.0
		-0.0638873	-0.1405521	0.0	0.0
			0.1851584	-0.1405521	0.0
				0.0544346	-0.1405521
					0.0851831

table 2.d

The r_i, $\bar{r}_i$, p_i and $\bar{p}_i$ are found on the diagonals of the matrices. Note that $r_i = \bar{p}_i$. Remark also that the $\mathbb{S}_i$ matrix is diagonal because it contains the numerators of row 0 in the Padé table i.e. constants. The $\bar{\mathbb{S}}_i$ matrix contains

the numerators of row 1, and therefore it is bidiagonal. The recursion coefficients are given in table 3.

i	α_i	β_i
0	1.6500000	0.21876933
1	4.2603593	-0.56368121
2	3.2340000	0.41590641
3	-7.1148000	-0.14055209

table 3

They can be found from the diagonal elements in table 2 according to their definitions

$$\alpha_i = -p_i/\bar{p}_i \qquad \text{and} \qquad \beta_i = -\bar{r}_i/r_{i-1}.$$

These coefficients can be used to give denominators $\mathbb{Q}_i$ of row 0 and denominators $\bar{\mathbb{Q}}_i$ of row 1 according to the recursion (2.14). The results are found in table 4.

i	4	3	2	1	0
$\mathbb{Q}_i$	1.0	1.65	4.2603593	3.23400000	-7.1148
		1.0	2.5820359	1.96000000	-4.3120
			1.0	0.75909091	-1.6700
				1.0	-2.2000
					1.0000

table 4.a

i	4	3	2	1	0
$\bar{\mathbb{Q}}_i$	1.0				
	0.60606060	1.0			
	0.23472199	1.06060606	1.0		
	0.309214595	0.51020408	-0.71130469	1.0	
	-0.14055209	0.41590641	-0.56368121	0.218769327	1.0

table 4.b

4. Recursions in the Laurent-Padé problem

At this stage, the Laurent-Padé approximation problem is introduced. We show that the algorithms from previous sections may be used to solve this problem. The Levinson and Schur algorithms will turn out to be special case of these.

Let be given

$$F(z) = \sum_{k=-\infty}^{\infty} f_k z^k$$

and define

$$F^{\pm}(z) = f_0 + 2 \sum_{k=1}^{\infty} f_{\pm k} z^k \quad ,$$

so that

$$F(z) = \frac{1}{2} [F^{+}(z) + F^{-}(1/z)] \ .$$

For nonnegative integers r and s, we call $\sum_{-r}^{s} \phi_j z^j$ a Laurent polynomial of degree max(r,s). An (m/n) Laurent-Padé approximant to F(z) is defined to be the following object (we drop the m/n indication for notational comfort)

$$[m/n] \triangleq (q^{-}(1/z))^{-1} (P(z)) (q^{+}(z))^{-1} \tag{4.1}$$

with

$$P(z) = \frac{1}{2} [q^{-}(1/z)p^{+}(z) + p^{-}(1/z)q^{+}(z)]$$

and where

(i) $q^{\pm}(z)$ are polynomials of degree n ,

(ii) $p^{\pm}(z)$ are polynomials of degree at most $\max(m,n)$,

(iii) $r^{+}(z) = (q^{+}(z))^{-1}(p^{+}(z))$ is a PA for $F^{+}(z)$,

$r^{-}(z) = (p^{-}(z))(q^{-}(z))^{-1}$ is a PA for $F^{-}(z)$,

(iv) $P(z)$ is of (Laurent) degree at most m.

Notice that for $m \geqslant n$, the Laurent-Padé problem reduces to just two regular Padé problems, since (iv) is then automatically satisfied. The more difficult case is when $m < n$. Then the Padé part of the problem determines only $m+n+1$ coefficients out of the $2n+1$. The extra conditions follow from the degree requirement of P in (iv).

Notice that in any case the Laurent series of $[m/n] = \frac{1}{2}[(p^{+}(z))(q^{+}(z))^{-1} + (q^{-}(1/z))^{-1}(p^{-}(1/z))]$ agrees with the given series $F(z)$ in the coefficients of z^{k} for $|k| < m+n+1$. We say that the residual $F(z) - [m/n]$ has Laurent order $m+n+1$.

It can be shown [8] that the conditions on the coefficients reduce to

$$\begin{array}{l|l} p_i^{+} = 2 \sum\limits_{j=0}^{i}{}' f_{i-j}\, q_j^{+} & p_i^{-} = 2 \sum\limits_{j=0}^{i}{}' q_j^{-}\, f_{j-i} \qquad , \ 0 \leqslant i \leqslant \ell = \max(m,n) \\ \sum\limits_{j=0}^{n} f_{i-j}\, q_j^{+} = 0 & \sum\limits_{j=0}^{n} q_j^{-}\, f_{j-i} = 0 \qquad , \ n < i \leqslant m+n \end{array} \tag{4.2}$$

The prime on the summation indicates that f_0 has to be replaced by $f_0/2$. Furthermore, $p_i^{\pm} = 0$ for $i < 0$ and $i > \ell$ and $q_j = 0$ for $j < 0$ and $j > n$.

It is clear that in (4.2) we have two Toeplitz systems, one for q^{+} and one for q^{-}. The left and right algorithms from the previous sections can be used

and there is no essential difference in the solution, except that now two instead of one Padé problem has to be solved.

To avoid all difficulties, we suppose again that $F(z)$ is normal, i.e. that all the Toeplitz matrices involved in the solutions of q^+ and q^- are nonsingular.

The Levinson-Schur algorithm will be related to row 0 of the Laurent-Padé table in a special case which is to be introduced now, together with some other applications of Laurent-Padé approximation [9].

1) Fourier-Padé approximation

Here we wish to approximate

$$S(\omega) = a_0 + 2 \sum_1^\infty (a_k \cos k\omega + b_k \sin k\omega), \quad a_k, b_k \in \mathbb{R}^{n\times n} .$$

This reduces to the Laurent-Padé problem as follows : set $z = e^{j\theta}$, $f_0 = a_0$ and $f_{\mp k} = a_k \pm j\, b_k$ $(k \geqslant 1)$, then

$$S(\omega) = F(z) = \sum_{-\infty}^{\infty} f_k z^k .$$

This Laurent series has the special property that f_k is the complex conjugate of f_{-k}.

2) Chebyshev-Padé approximation

The problem is as above but now $b_k = 0$ for $k \geqslant 1$. So the series reduces to a Chebyshev series (of the first kind) in x, if we set $x = \cos\omega$. Moreover $f_k = f_{-k}$ is real. Chebyshev series of the second kind are treated in a similar way.

3) Trigonometric moment problem

In this case, the f_k are the moments of an Hermitian valued measure of the unit circle

$$f_k = \frac{1}{2\pi} \int_{-\pi}^{\pi} e^{-jk\theta}\, d\omega(\theta).$$

This implies that $f_{-k} = f_k^{\star}$, where upper star indicates the adjoint.

For a relation to the two-point Padé approximation see [10].

The symmetry in the coefficients may be exploited in the last case. Only one Padé approximation has to be computed, the other being identical or simply the adjoint. Indeed one can easily verify by inspecting (4.2) that if $f_{-k} = f_k^{\star}$, then

$$q^+(z) = (q^-(z^{\star}))^{\star} \qquad \text{and} \qquad p^+(z) = (p^-(z^{\star}))^{\star} \tag{4.3}$$

up to some normalization.

At this point we introduce the parahermitian conjugate which is defined by :

$$f_{\star}(z) = (f(1/z^{\star}))^{\star}.$$

Thus $q^-(1/z) = q_{\star}^+(z)$, $p^-(1/z) = p_{\star}^+(z)$ and $r^-(1/z) = r_{\star}^+(z)$.
Hence, all quantities may be expressed in terms of p^+ and q^+. We have e.g. that for 2) or 3)

$$[m/n] = (q_{\star}^+(z))^{-1}\, \frac{1}{2}[q_{\star}^+(z)p^+(z) + p_{\star}^+(z)q^+(z)](q^+(z))^{-1} \tag{4.4a}$$

$$= \frac{1}{2}\,[(p^+(z))(q^+(z))^{-1} + (q_{\star}^+(z))^{-1}(p_{\star}^+(z))] \tag{4.4b}$$

If it were possible to factor P(z), which is the central factor in (4.4a), then it would be possible to find the spectral factors of F(z). However, P(z) is not positive semi-definite in general, except when m = 0. Only in that case

is [m/n] a nonnegative real function on the unit circle if F(z) is a nonnegative real function. This fact is best illustrated in the scalar case. It is easy to verify that if β is a zero of P(z), then $1/\beta^*$ is also a zero of P(z), however these coincide if $|\beta| = 1$. So a spectral factorization of P(z) is only possible if all unimodular zeros are of even multiplicity. We illustrate this by an example.

Example 2

Take the low-pass filter of [4], i.e. $F(z^{-1}) = f_0 + 2 \sum_{k=1}^{\infty} f_k z^{-k}$ with

$f_0 = (\omega_s+\omega_p)/2\pi$, $f_k = \frac{1}{(\omega_s-\omega_p)k^2} [\cos(k\omega_p)-\cos(k\omega_s)]$

where $\omega_s = 0.5201\pi$ and $\omega_p = 0.4299\pi$.

The Laurent-Padé table has been computed for $0 \leq m, n \leq 6$.

Zeros not on the unit circle appear only for $m \geq n+2$.

But all entries for $m \geq 1$ have simple zeros about equally spaced on the left half of the unit circle. This was to be expected since the approximation oscillates around the exact value in the stopband, as illustrated in appendix 3.

We remark also that $q^+(z)$ as defined above has zeros which are usually stable. Only the (2,1), (4,1), (4,3), (6,1), (6,4) and (6,6) entries have an unstable pole from $q^+(z)$, but in those cases they are approximately cancelled by a corresponding zero.

5. The Levinson and Schur algorithms

Now we pass to the digital filter application, where the Levinson and Schur algorithms were first formulated. We show that they coincide with the previous algorithms in a special case. Note that in this section we adopt the engineering convention to work with series in z^{-1}, as opposed to the theory in previous sections.

Suppose first that an outer function $\Sigma(z) = \sum_{k=0}^{\infty} s_k z^{-k} \in M$ is given which is the minimal phase transfer function of a stable filter, i.e. all transmission zeros are outside or on the unit circle. A padé approximant of $\Sigma(z)$ will usually not have the same properties.

We illustrate this again by a scalar example.

Example 3

Let $\Sigma(z) = [(z^{-1} + 1.1)(z^{-1} - 3.3)(z^{-2} + 1.96)]^{-1}$

with poles given in table 5

i	pole i of S(z)
1	$-1/1.1 \simeq -0.90909$
2	$1/3.3 \simeq 0.30303$
3	$j/1.4 \simeq 0.71429j$
4	$-j/1.4 \simeq -0.71429j$

table 5

All poles are inside the open unit disc, which stands for stability, and four zeros are at infinity.

The power series expansion $\Sigma(z) = \sum_{k=0}^{\infty} s_k z^{-k}$ has coefficients given in table 6.

k	s_k
0	-0.1406e+00
1	-0.8518e-01
2	-0.1864e-01
3	0.3476e-01
4	-0.6279e-01
5	0.4763e-01

table 6

The (1,2) entry of the Padé table is

$$[1/2] = \frac{p_0+p_1 z^{-1}}{q_0+q_1 z^{-1}+q_2 z^{-2}} \quad ,$$

the coefficients of which are resumed in table 7, together with zero a_0 and the poles b_0 and b_1.

i	p_i	q_i	a_i	b_i
0	0.24934677	-1.7740524	1.317	1.186
1	-0.32848078	1.2618918	-	0.475
2	-	1.0	-	-

table 7

Clearly b_0 is an unstable pole.

Another approach to the problem is to find the transfer function as a spectral factor of the function $C(z) = \Sigma(z)\Sigma_\star(z)$. Thus suppose that we know the covariance coefficients c_k, i.e. we know the power spectrum

$$C(e^{j\omega}) = \Sigma(e^{j\omega})\Sigma(e^{j\omega})^\star = c_0 + 2\sum_{k=1}^{\infty} c_k \cos k\omega \; . \qquad (5.1)$$

Define

$$F(z) = c_0 + 2\sum_{k=1}^{\infty} c_k z^{-k}$$

then

$$C(z) = \frac{1}{2}\left[F(z) + F_\star(z)\right] \; .$$

The problem is now : given this information, find an approximation $\Sigma_a(z)$ for $\Sigma(z)$ such that their power spectra match as well as possible, i.e.

$$\Sigma(z)\Sigma_\star(z) \approx \Sigma_a(z)\Sigma_{a\star}(z).$$

Therefor we first try to find a Laurent-Padé approximation for $C(\omega)$ and then look for its spectral factor which is a candidate for $\Sigma_a(z)$. Because $c_k = c^\star_{-k}$, we are in the case of a Laurent-Padé approximation for the trigonometric moment problem, and we can use the form (4.4) for [m/n]. Unfortunately, the spectral factorization is not always possible, as illustrated in example 2, except if we restrict ourselves to row zero, i.e. [0/n], n = 0,1,2,... . Take the system for q^+ from (4.2) and add a row for i = 0 as we did in (2.6) but with a comonic normalization for q^+. This yields an hermitean matrix for the system

$$T(0,n,n)\begin{bmatrix} I \\ q_1^+ \\ \vdots \\ q_n^+ \end{bmatrix} = \begin{bmatrix} \bar{p}_0 \\ 0 \\ \vdots \\ 0 \end{bmatrix} . \qquad (5.2)$$

Because C(z) is a power spectrum, we know that $\{c_k\}$ is a nonnegative sequence and thus all T(0,n,n) are positive definite and well conditioned. Because of the normalization for q^+, we identify it with the $\bar{Q}_n$ from (2.14), (2.15) or (2.16), by which we can solve the system (5.2) if we take m =-1. In that case is $\bar{Q}_n$ the right denominator of [0/n]. A possible scheme in this case is (2.16). We start with $r_0 = p_0 = f_0$. $q^+ = \bar{Q}_n$ is the forward predictor and Q_n is the backward predictor. They represent the Szegö polynomials for the multivariable case [17]. α_n and β_n are the partial correlation coefficients or reflection coefficients. Symmetry considerations lead to the observation that $p_n = \bar{r}^\star_n$ [11,12]. The relation with the left denominator and q^- is that $_n\bar{Q}^\star = \bar{Q}^t_n = [I (q_1^-)^\star \ \ldots \ (q_n^-)^\star]$. We mention also that we could normalize $Q(z)$ and $\bar{Q}(z)$

(polynomials in z^{-1}) such that we allow another unitary matrix instead of I for their leading or constant coefficient. If this is done correctly (see appendix 2), then (2.17) reduces to

$$[Q_{n+1}(z)\bar{Q}_{n+1}(z)] = [Q_n(z)\bar{Q}_n(z)] \begin{bmatrix} z^{-1}I & 0 \\ 0 & I \end{bmatrix} \begin{bmatrix} I & \rho_n^* \\ \rho_n & I \end{bmatrix} \begin{bmatrix} v_n^{-1} & 0 \\ 0 & w_n^{-1} \end{bmatrix}$$

$$= [Q_n(z)\bar{Q}_n(z)]\tilde{\theta}_n \qquad (5.3)$$

with $v_n = (I - \rho_n^* \rho_n)^{R/2}$ and $w_n = (I - \rho_n \rho_n^*)^{R/2}$.

$R/2$ indicates an appropriate right square root. Note that these exist because $\|\rho_n\| \leq 1$ where spectral norm is meant.

This is the normalized form of the Levinson-Wiggins-Robinson scheme, which is of incoming type. The outgoing analog or Schur method follows from (3.5) with $m = -1$ and is given in its condensed form by (3.6), i.e.

$$\begin{bmatrix} R_{n+1}(z)\bar{R}_{n+1}(z) \\ S_{n+1}(z)\bar{S}_{n+1}(z) \end{bmatrix} = \begin{bmatrix} R_n(z)\bar{R}_n(z) \\ S_n(z)\bar{S}_n(z) \end{bmatrix} \tilde{\theta}_n \qquad (5.4)$$

where $\tilde{\theta}_n$ is as in (5.3). $R_n(z)$ and $\bar{R}_n(z)$ are as defined in section 3 and they represent residual series, i.e. formal power series in z^{-1}. $S_n(z)$ and $\bar{S}_n(z)$ are also as defined in section 3 but because now f_k is not vanishing for negative k, these are not polynomials of finite degree, but are generalizations thereof, being infinite formal power series in z. I.e.

$$S_n(z) = [\ \ldots\ z^2 I \quad zI]\, \mathbb{S}_n$$
$$\bar{S}_n(z) = [\ \ldots\ zI \quad I]\, \mathbb{S}_n \quad . \qquad (5.5)$$

The resulting approximation of $C(z) = \Sigma(z)\Sigma_\star(z)$ is then constructed from

$$q^+(z) = [I \quad z^{-1}I \ \ldots \ z^{-n}I]\, \bar{Q}_n = q_\star^-(1/z)$$

and

$$P(z) = \frac{1}{2}\left[p^+(z)(q^+(z))^{-1} + (q_\star^+(z))^{-1}(p_\star^+(z))\right]$$

which is a constant positive definite matrix. Let t be a right square root of P, then

$$\Sigma_a(z) = t(q^+(z))^{-1}$$

approximates $\Sigma(z)$ in a least squares sense [17].
$q^+(z)$ is a polynomial in z^{-1} with zeros outside the unit disc, so that stability is guaranteed.

Looking at (2.10) - (2.13) we see that the incoming and outgoing algorithms give the block triangular factorizations of block Toeplitz matrices. The recursion however doesn't allow for pivoting to maintain numerical accuracy. Therefore it is clear that recursive techniques to compute Padé approximations are numerically instable in general. However, in the case of Levinson-Schur algorithms, we have positive definite Toeplitz matrices. These can be factorized in a numerically stable way by the Cholesky method without pivoting. A similar property seems to be shared by the Schur and Levinson schemes [13].

The elementary matrices $\tilde{\theta}_i$ have also their physical interpretation. They characterize the elementary sections of a ladder filter. I.e. the ρ_i are reflection coefficients for a discrete analog of a wave propagating system. See e.g. [14-16].

Remark that only the AR scheme works well in general, i.e. Σ_a can be found because the Laurent-Padé approximation can be factorized. The ARMA case can be treated if we leave the Padé scheme and construct a Laurent-Hermite inter-

polant for the given autocorrelation function. This is a so called Nevanlinna-Pick problem which is treated elsewhere [18,16].

References

1. F. Brophy - A.C. Salazar, "Considerations of the Padé approximant technique in the synthesis of recursive digital filters", IEEE Trans. Audio Electroacoustics, vol. AU-21, pp. 500-505, 1973.

2. ____, "Recursive digital filter synthesis in the time domain", IEEE Trans. Acoust., Speech, Signal Processing, vol. ASSP-22, pp. 45-55, 1974.

3. C.K. Chui, P.W. Smith, L.Y. Su, "A minimization problem related to Padé synthesis of recursive digital filters" in Padé and rational approximation theory and applications, E.B. Saff and R.S. Varga (eds.), New York, Acad. Press, 1977, pp. 247-256.

4. C.K. Chui, A.K. Chan, "A two-sided rational approximation method for recursive digital filtering", IEEE Trans. Acoust., Speech, Signal Proc., vol. ASSP-27, pp. 141-145, 1979.

5. R. Hastings-James, S. Mehra, "Extensions of Padé approximant technique for the design of recursive digital filters", IEEE Trans. Acoust., Speech, Signal Proc., vol. ASSP-25, pp. 501-509, 1977.

6. D. Bessis, "Topics in the theory of Padé approximants" in P.R. Graves-Moris (ed.) Padé approximants. The Institute of Physics, London-Bristol, 1973, pp. 19-44.

7. A. Bultheel, "Recursive algorithms for the matrix Padé problem", Math. of Comp., vol. 35, 151, July 1980.

8. W.B. Gragg, G.D. Johnson, "The Laurent-Padé table" in Information processing 74, Proc. IFIP congress, 1975, North-Holland, Amsterdam 1974, pp. 632-637.

9. W.B. Gragg, "Laurent, Fourier and Chebyshev-Padé tables" in E.B. Saff, R.S. Varga (eds.), Padé and rational approximation, theory and application, Academic Press, New York, 1977, pp. 61-72.

10. W.J. Thron, "Two-point Padé tables, T-fractions and sequences of Schur" in E.B. Saff, R.S. Varga (eds.), Padé and rational approximation, theory and applications, Academic Press, New York, 1977, pp. 215-226.

11. H. Akaike, "Block Toeplitz matrix inversion", SIAM J. Appl. Math., vol. 24, 1973, pp. 234-241.

12. J. Rissanen, "Algorithms for triangular decomposition of block Hankel and Toeplitz matrices, applications to factoring positive matrix polynomials", Math. of Comp., vol. 27, 1973, pp. 147-154.

13. G. Cybenko, "Round-off error propagation in Durbin's, Levinson's and Trench's algorithms", presented at Internatl. conf. on Acoust., Speech and Signal Proc., IEEE, Washington DC, April 1979.

14. H. Wakita, "Direct estimation of the vocal tract shape by inverse filtering of acoustic speech wave forms", IEEE Trans. on audio electro-acoustics, AU-21, 1973, pp. 417-427.

15. E.A. Robinson, S. Treitel, "The spectral function of a layered system and the determination of the waveforms at depth", Geophysical prospecting, vol. 25, 3, 1977, pp. 434-459.

16. P. Dewilde, A. Vieira, T. Kailath, "On a generalized Szegö-Levinson realization algorithm for optimal linear predictors based on a network synthesis approach", IEEE Trans. on Circuits and Systems, vol. CAS-25, 1978, pp. 663-675.

17. P. Delsarte, Y. Genin, Y. Kamp, "Orthogonal polynomial matrices on the unit circle", IEEE Trans. on Circuits and Systems, vol. CAS-25, 1978, pp. 149-160.

18. A. Bultheel, "Recursive rational approximations", Ph. D. Thesis, K.U.Leuven, Sept. 1979 (in Dutch).

Appendix 1

The bordering technique for the derivation of the recursions is a very simple one. As an example we derive some of the recursions given in section 2. Keeping in mind that the last element of $\mathbb{Q}_i$ is I and that $\mathbb{Q}_{n+1}$ must satisfy

$$T_{n+1}\,\mathbb{Q}_{n+1} = [0 \ \ldots \ 0 \ \star]^t \tag{A1.1}$$

we can use the fact that, due to the Toeplitz structure of T_{n+1} we have

$$T_{n+1}\left[\begin{array}{c|c|c} 0 & 0 & \mathbb{Q}_n \\ & \mathbb{Q}_{n-1} & \\ \mathbb{Q}_n & 0 & 0 \end{array}\right] = \left[\begin{array}{c|c|c} p_n & p_{n-1} & 0 \\ 0 & 0 & \vdots \\ \vdots & \vdots & \vdots \\ 0 & 0 & 0 \\ 0 & r_{n-1} & r_n \\ r_n & r'_{n-1} & r'_n \end{array}\right] \tag{A1.2}$$

so that elimination of undesired nonzero elements on the right disappear by taking the linear combination given by the recursion (2.9). This is the only possible one which preserves the normalization.

The recursion (2.14) can be found in a similar way. First we must have that $\mathbb{Q}_{n+1}$ satisfies (A1.1) while we have that

$$T_{n+1}\left[\begin{array}{c|c} 0 & \bar{\mathbb{Q}}_n \\ \mathbb{Q}_n & 0 \end{array}\right] = \left[\begin{array}{c|c} p_n & \bar{p}_n \\ 0 & 0 \\ \vdots & \vdots \\ 0 & 0 \\ r_n & \bar{r}_n \end{array}\right] . \tag{A1.3}$$

The first recursion in (2.14) gives the combination of $\mathbb{Q}_n$ and $\bar{\mathbb{Q}}_n$ to let (A1.1) happen together with the proper normalization for $\mathbb{Q}_{n+1}$. By upper baring (A1.1) we obtain the requirement for $\bar{\mathbb{Q}}_{n+1}$ with comonic normalization, i.e. the first

element of $\bar{Q}_i$ has to be I. Note now that

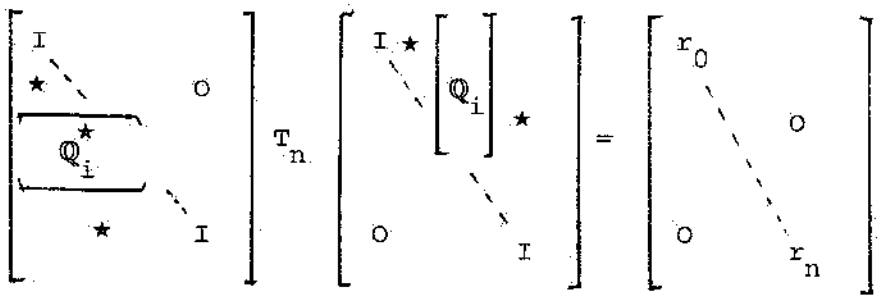
$$\bar{T}_{n+1} \left[\begin{array}{c|c} & \bar{Q}_n \\ Q_{n+1} & \\ & 0 \end{array}\right] = \left[\begin{array}{c|c} 0 & 0 \\ \vdots & \vdots \\ 0 & 0 \\ r_{n+1} & \bar{r}_n \\ r'_{n+1} & \bar{r}'_n \end{array}\right] \qquad \text{(A1.4)}$$

so that $Q_{n+1} - [\bar{Q}_n^t \;\; 0]^t \bar{r}_n^{-1} r_{n+1}$ meets the requirement for $\bar{Q}_{n+1}$ except for the normalization. The first element is indeed $\nu_n^{-1} = q_0^{(m/n+1)} - \tilde{r}_n^{-1} r_{n+1}$, so that multiplying with ν_n as in (2.14) gives the proper normalization. (2.15) can be derived in exactly the same way.

The first equations of (2.14) and (2.15) give (2.16).

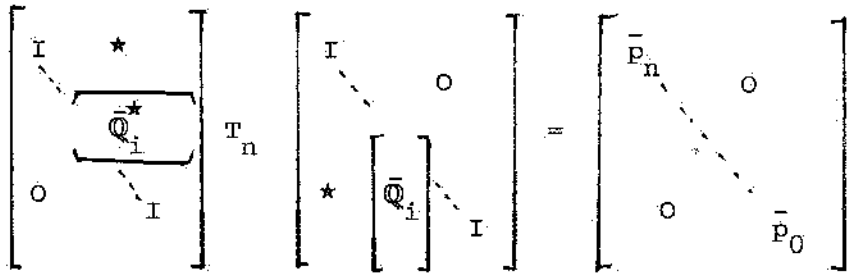
Appendix 2

It follows from (2.10) and (2.13) that



(A2.1)

and



(A2.2)

which indicates the orthogonality of $\mathbb{Q}_i$ and $\bar{\mathbb{Q}}_i$ with respect to the Toeplitz weight. Since the right hand side is not the unit matrix, the corresponding polynomials are not normal in a strict sense, i.e.

$$\mathbb{Q}_n^{\star} T_n \mathbb{Q}_n = r_n \neq I \ .$$

Define the normalized quantity

$$\mathbb{Q}_{\mathfrak{n}} = \mathbb{Q}_n (r_n^{R/2})^{-1}$$

where $r_n^{R/2}$ is an arbitrary right square root of r_n. This leaves $\mathbb{Q}_{\mathfrak{n}}$ undefined for a right unitary factor, but this is unavoidable. Similarly we can set

$$\bar{\mathbb{Q}}_{\mathfrak{n}} = \bar{\mathbb{Q}}_n (\bar{p}_n^{R/2})^{-1} \ .$$

Substitution into (2.16) gives the recursions for the normalized denominators :

$$\mathbb{Q}_{n+1} = ([0 \quad \mathbb{Q}_n^t]^t + [\bar{\mathbb{Q}}_n^t \quad 0]^t \; \mu_i) r_i^{R/2} (r_{i+1}^{R/2})^{-1} ,$$

$$\mu_i = \bar{p}_i^{R/2} \; \alpha_i (r_i^{R/2})^{-1} = -((\bar{p}_i^{-R/2})^{\star})^{-1} \; p_i (r_i^{R/2})^{-1}$$

$$\bar{\mathbb{Q}}_{n+1} = ([\bar{\mathbb{Q}}_n^t \quad 0]^t + [0 \quad \mathbb{Q}_n^t]^t \; \nu_i) \bar{p}_i^{-R/2} (p_{i+1}^{R/2})^{-1} ,$$

$$\nu_i = r_i^{R/2} \; \beta_i (\bar{p}_i^{R/2})^{-1} = -((r_i^{R/2})^{\star})^{-1} \; \bar{r}_i (\bar{p}_i^{R/2})^{-1} .$$

Now because $p_i^{\star} = \bar{r}_i$ we also have $\mu_i = \nu_i^{\star}$.

Furthermore $r_{i+1} = r_i (I - \beta_i \alpha_i)$ gives, when introducing μ_i :

$$((r_i^{R/2})^{\star})^{-1} (r_{i+1}^{R/2})^{\star} (r_{i+1}^{R/2}) (r_i^{R/2})^{-1} = (I - \mu_i^{\star} \mu_i).$$

This implies that we may take $r_{i+1}^{R/2} (r_i^{R/2})^{-1} = (I - \mu_i^{\star} \mu_i)^{R/2}$.

Similarly one can take $\bar{p}_{i+1}^{-R/2} = (I - \mu_i \mu_i^{\star})^{R/2} \; \bar{p}_i^{-R/2}$.

To fix the constant unitary factor we can chose it such that $q_n^{(m/n)}$ is an hermitian unitary matrix. Then we have to take the hermitian square roots of the defect matrices $I - \mu_i^{\star} \mu_i$ and $I - \mu_i \mu_i^{\star}$ so that the algorithm reduces in this case to (hermitian square root is indicated by $^{1/2}$)

$$r_0^{1/2} = \mathbb{Q}_0 = c_0^{1/2} - \bar{\mathbb{Q}}_0 = p_0^{-1/2}$$

for $i = 0,1,2,\ldots$

$$\mathbb{Q}_{i+1} = \left(\begin{bmatrix} 0 \\ \mathbb{Q}_i \end{bmatrix} + \begin{bmatrix} \bar{\mathbb{Q}}_i \\ 0 \end{bmatrix} \rho_i \right) v_i \qquad \bar{\mathbb{Q}}_{i+1} = \left(\begin{bmatrix} \bar{\mathbb{Q}}_i \\ 0 \end{bmatrix} + \begin{bmatrix} 0 \\ \mathbb{Q}_i \end{bmatrix} \rho_i^{\star} \right) w_i$$

$$r_{i+1}^{1/2} = (I - \rho_i^{\star} \rho_i)^{1/2} \; r_i^{1/2} \qquad \bar{p}_{i+1}^{-1/2} = (I - \rho_i \rho_i^{\star})^{1/2} \; \bar{p}_i^{-1/2}$$

with

$$-\bar{p}_i^{-1/2}\,[c_{-1}\ \cdots\ c_{-i-1}]\,\bar{Q}_i = \rho_i = (r_i^{-1/2}\,[c_{i+1}\ \cdots\ c_1]\,\bar{Q}_i)^*$$

$$v_i = (I - \rho_i^*\rho_i)^{-1/2} \qquad w_i = (I - \rho_i\rho_i^*)^{-1/2}\ .$$

Appendix 3

In this appendix we represent graphically the power spectra of the Laurent-Padé approximations [m/n] to the lowpass filter considered in [4].

The power spectrum is given by

$$H(\ ) = f_0 + 2 \sum_{k=1}^{\infty} f_k \cos k\omega$$

with $f_0 = (\omega_s + \omega_p)/2\pi$

and $f_k = [\cos(k\omega_p) - \cos(k\omega_s)]/((\omega_s-\omega_p)k^2\pi)$,

$\omega_s = 0.5201\pi$, $\omega_p = 0.4299\pi$

(remark that in [4] the factor 2 before the summation has been forgotten and that two digits of ω_p and ω_s have been switched).

The Laurent-Padé approximations [m/n] for

$$f(z) = \sum_{-\infty}^{\infty} f_k z^k$$

are computed for $0 \leqslant m,n \leqslant 6$ and the resulting approximants $H_L(\omega)$ are plotted for $0 \leqslant \omega \leqslant \pi$ in the next figures. They are grouped row by row, each row on one page.

The scale in y-direction is different for each function, therefore we have crossed the points (0,0.5) and (0,1) on the y-axis to indicate the scale used. Remark that row zeros, which gives the Levinson approximants, yields positive real approximations for each value of n but the approximation is not good in the passband.

The zeros on the unit circle are found as points where the approximation intersects the x-axis in the stopband. For higher values of n, the approximation in the stopband is quite good, but the approximation is not positive real (there are intersections with the x-axis, invisible at the scale used). It is clear that if we add a very small constant to the given power spectrum, then the approximation will probably be positive real and deviate only very little from the Laurent-Padé approximation of the exact spectrum.
Remark however that adding a constant to the given power series does not result in adding the same constant to the Laurent-Padé approximant.