A look-ahead algorithm for the solution of block Toeplitz systems*

Marc Van Barel and Adhemar Bultheel

Department of Computer Science
Katholieke Universiteit Leuven
Celestijnenlaan 200 A
B-3001 Heverlee, Belgium
e-mail: Marc.VanBarel@cs.kuleuven.ac.be

Submitted by Georg Heinig

ABSTRACT

In this paper, we give a weakly stable algorithm to solve a block Toeplitz system of linear equations. If the "look-ahead" steps taken to compute the parameters of the inversion formula for the block Toeplitz matrix are small compared to the order n of the matrix, the algorithm requires $O(n^2)$ floating point operations. The parameters of the inversion formula are interpreted and computed in a recursive way as solutions of certain interpolation problems given the formal Laurent series based on the data of the block Toeplitz matrix.

short running title: block Toeplitz systems

1. Introduction

In this paper, we look for the solution of a system of linear equations

$$T_{m,n} V_{m,n} = Y_{m,n} \tag{1}$$

where $T_{m,n}$ is a block Toeplitz coefficient matrix, i.e. the elements are equal along each block diagonal

$$T_{m,n} \quad = \quad [t_{i-j}]_{i=0,1,\ldots,m-1}^{j=0,1,\ldots,n-1}$$

* This research was supported by the National Fund for Scientific Research (NFWO), project Lanczos, grant #2.0042.93 and by the Human Capital and Mobility project ROLLS of the European Community under contract ERBCHRXCT930416.

© Elsevier Science Publishing Co., Inc., 199.
655 Avenue of the Americas, New York, NY 10010 0024-3795/92/$5.00

$$= \begin{bmatrix} t_0 & t_{-1} & \cdots & t_{-n+2} & t_{-n+1} \\ t_1 & t_0 & \cdots & t_{-n+3} & t_{-n+2} \\ \vdots & \vdots & & \vdots & \vdots \\ t_{m-1} & t_{m-2} & \cdots & t_{m-n+1} & t_{m-n} \end{bmatrix}$$

where the entries t_i are $p \times q$ real or complex matrices. Moreover, $T_{m,n}$ is square, i.e., $pm = qn$.

The classical algorithms to solve (1) exploit the Toeplitz structure and require only $O((pm)^2)$ operations, compared to $O((pm)^3)$ for general linear systems. The flow of these methods is determined by the exact singularity of the square leading principal block submatrices of $T_{m,n}$. They compute the solutions for successive nonsingular leading principal submatrices. However, in finite-precision arithmetic it turns out that also the leading principal submatrices which are nonsingular but ill-conditioned should be avoided. For scalar Toeplitz matrices, several of these look-ahead algorithms have been designed [8, 7, 10, 9, 19, 21, 20, 26, 12, 18]. Some of these are even superfast, i.e. requiring $O(m \log^2 m)$ operations [19, 21, 20]. Also for Hankel matrices, i.e. matrices having the same entries on each antidiagonal, such look-ahead schemes have been constructed [6, 11, 1]. Note that the Hankel case is simpler than the Toeplitz case because Hankel matrices are symmetric and when translating these problems into series, the Hankel problem can be translated using formal power series while for the Toeplitz case we need formal Laurent series. For the Hankel case, [6] gives the error analysis. As far as we know, such a detailed error analysis was not done for the look-ahead procedures to solve the (scalar) Toeplitz problem. In this paper, we want to design a look-ahead algorithm solving the block Toeplitz system (1) and motivate the look-ahead strategy by the results of a detailed error analysis. In [14], a look-ahead Schur algorithm is designed for Hermitian block Toeplitz matrices. Several high performance algorithms for Toeplitz and block Toeplitz matrices are described in [13] including 2 look-ahead Schur algorithms for symmetric indefinite block Toeplitz matrices.

A totally different recent approach to solve Toeplitz systems is based on several possible transformations into a generalized Cauchy matrix. Such a generalized Cauchy matrix can be factorized using pivoting without destroying the structure. Hence, without using look-ahead the Cauchy system can be solved and the solution can be transformed back to the solution of the Toeplitz system. We refer the interested reader to [15, 27, 23, 24, 25, 25, 13] and the references therein.

In [4, 2, 3], Cabay et al. give the error analysis for a "look-ahead" algorithm computing multi-dimensional Padé systems and inverting generalized Sylvester matrices. In a similar way, we want to give the error analysis for the inversion of block Toeplitz matrices and the solution of the

related interpolation problems. I.e., we follow the same path of reasoning as in [4, 2, 3], but the content will be adapted to block Toeplitz instead of generalized Sylvester matrices. This change of content turns out to be nontrivial, especially for finding the corresponding interpolation problems and the updating formulas for the parameters of the inversion formula of the block Toeplitz matrix. The nontrivial part of the adaptation of the results of [4, 2, 3] is described in this paper, while we only mention the main results for the trivial part and refer for the details to our report [28]. To get a similar flow of the error analysis as in [4, 2, 3], we derive the "look-ahead" algorithm in a specific way. Other implementations of the updating formulas are possible leading to another error analysis. In [5] a FORTRAN implementation of the algorithm designed in [4, 3] is described and the results of numerical experiments are given.

Section 2 shows the inversion formula which we will use for a block Toeplitz matrix. Section 3 shows how the parameters of this inversion formula can be interpreted as solutions of a right and a left or dual homogeneous interpolation problem. A so-called *basis matrix* (some authors also use the term *generating system*) allows to represent all solutions of each interpolation problem in a compact way. In Section 4, we develop an updating procedure for this basis matrix. In the error analysis, we need a connection between the basis matrices of the original and the dual interpolation problem. This duality is given for so-called *extended* basis matrices in Section 5. Hence, instead of working with the original basis matrices, we derive in Section 7 an updating algorithm for these extended basis matrices. This updating algorithm uses norms which are defined in Section 6. Based on the computed inversion parameters, Section 9 gives formulas for the inverse of the block Toeplitz matrix. Section 10 shows that the "look-ahead" algorithm computes the basis matrix in a weakly stable way. In Section 11, we indicate that using the inversion formula to compute the solution of a block Toeplitz system of linear equations is weakly stable. Section 12 gives some experimental results. The look-ahead strategy requires the user to input a threshold value. In Section 13, we design another look-ahead strategy where the user has only to give the maximum size of the look-ahead step taken. The experimental results, however, are less accurate than the results given by Freund and Zha in [10] for scalar Toeplitz matrices. It turns out that using the inversion formula leads to this loss of accuracy. Therefore, we constructed the updating formula of Section 14 for the solution of the block Toeplitz system of linear equations. In Section 15, we refine the look-ahead strategy to get experimental results which are now comparable to those of Freund and Zha for scalar matrices. We derive even better results when applying one step of iterative refinement using the inversion formula as was done in [21]. We also apply the procedure to 100 *block* Toeplitz matrices generating acceptable results and indicating

that this last look-ahead procedure together with the recursive updating of the solution or using iterative refinement is weakly stable. However, in some rare cases there is some greater loss of accuracy. For the moment, the mechanism behind this is not clear. In Section 16, we give the main result of the error analysis for the recursive updating formula. We tried to incorporate this result into another look-ahead criterion not only taking into account the error in computing the basis matrices but also the error in using the recursive updating. In some cases this gives a better result, but generally the result is a little bit worse than the look-ahead criterion of Section 15.

2. Inversion formula

In this section, we give an inversion formula for a block Toeplitz matrix. The parameters of this inversion formula will be interpreted as solutions of certain interpolation problems in Section 3. We repeat the result of Gohberg and Shalom [16, Th. 6.1].

THEOREM 1. *Let T be a square block Toeplitz matrix of the form $T_{m,n} = T = [t_{i-j}]_{i=0,1,\ldots,m-1}^{j=0,1,\ldots,n-1}$ with entries $t_k \in \mathbb{R}^{p\times q}$ and $mp = nq$.**

The matrix T is invertible if (and only if) there exist solutions to the following systems of matrix equations:

$$\begin{bmatrix} t_0 & t_{-1} & \cdots & t_{-n+1} & t_{-n} \\ t_1 & t_0 & \cdots & t_{-n+2} & t_{-n+1} \\ \vdots & \vdots & & \vdots & \vdots \\ t_{m-1} & t_{m-2} & \cdots & t_{m-n} & t_{m-n-1} \end{bmatrix} \begin{bmatrix} x_0 & -z_0 \\ x_1 & -z_1 \\ \vdots & \vdots \\ x_{n-1} & -z_{n-1} \\ 0 & I_q \end{bmatrix} = \begin{bmatrix} I_p & 0 \\ 0 & 0 \\ \vdots & \vdots \\ 0 & 0 \\ 0 & 0 \end{bmatrix} \tag{2}$$

and

$$\begin{bmatrix} 0 & u_{m-1} & \cdots & u_1 & u_0 \\ I_p & -w_{m-1} & \cdots & -w_1 & -w_0 \end{bmatrix} \begin{bmatrix} t_{-1} & t_{-2} & \cdots & t_{-n+1} & t_{-n} \\ t_0 & t_{-1} & \cdots & t_{-n+2} & t_{-n+1} \\ \vdots & \vdots & & \vdots & \vdots \\ t_{m-1} & t_{m-2} & \cdots & t_{m-n+1} & t_{m-n} \end{bmatrix} = \begin{bmatrix} 0 & 0 & \cdots & 0 & I_q \\ 0 & 0 & \cdots & 0 & 0 \end{bmatrix}$$

*The results developed in this paper for real numbers can be easily adapted to complex numbers.

with $x_j \in \mathbb{R}^{q\times p}$, $z_j \in \mathbb{R}^{q\times q}$, $u_j \in \mathbb{R}^{q\times p}$ *and* $w_j \in \mathbb{R}^{p\times p}$. *Note that* $z_0u_0 = x_0w_0$. *Moreover,* T^{-1} *admits the following representations:*

$$\begin{aligned} T^{-1} &= L_{(n,\eta)}(x_0, x_1, \ldots, x_{n-1})U_{(\eta,m)}(I_p, -w_{m-1}, -w_{m-2}, \ldots, -w_1) + \\ &\quad L_{(n,\eta)}(z_0, z_1, \ldots, z_{n-1})U_{(\eta,m)}(0, u_{m-1}, u_{m-2}, \ldots, u_1) \end{aligned}$$

and

$$\begin{aligned} T^{-1} &= U_{(n,\eta)}(I_q, -z_{n-1}, -z_{n-2}, \ldots, -z_1)L_{(\eta,m)}(u_0, u_1, \ldots, u_{m-1}) + \\ &\quad U_{(n,\eta)}(0, x_{n-1}, x_{n-2}, \ldots, x_1)L_{(\eta,m)}(w_0, w_1, \ldots, w_{m-1}) \end{aligned}$$

for $\eta = \min(m, n)$. *Here, for* $k \geq l$,

$$L_{(k,l)}(g_0, g_1, \ldots, g_{k-1}) = \begin{bmatrix} g_0 & 0 & \cdots & 0 \\ g_1 & g_0 & \ddots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ \vdots & \vdots & & g_0 \\ \vdots & \vdots & & \vdots \\ g_{k-1} & g_{k-2} & \cdots & g_{k-l} \end{bmatrix},$$

and for $k \leq l$,

$$L_{(k,l)}(g_0, g_1, \ldots, g_{l-1}) = \begin{bmatrix} g_{l-k} & \cdots & g_0 & 0 & \cdots & 0 \\ g_{l-k+1} & \cdots & g_1 & g_0 & \cdots & 0 \\ \vdots & & \vdots & \vdots & & \vdots \\ g_{l-1} & \cdots & g_{k-1} & g_{k-1} & \cdots & g_0 \end{bmatrix}.$$

Similarly, for $k \geq l$, $U_{(k,l)}(g_0, g_1, \ldots, g_{k-1}) = L_{(l,k)}(g_0^T, g_1^T, \ldots, g_{k-1}^T)^T$ *and for* $k \leq l$, $U_{(k,l)}(g_0, g_1, \ldots, g_{l-1}) = L_{(l,k)}(g_0^T, g_1^T, \ldots, g_{l-1}^T)^T$.

Proof. This theorem can be obtained as a special result of Theorem 18 when the computations are considered to be done with infinite precision, i.e. $\theta_{VIII} = 0$. ■

Note that for the scalar case, i.e., when $p = q = 1$, it follows that $m = n$, $x_j = u_j$ and $z_j = w_j$ for $j = 0, 1, \ldots, n-1$.

The couple $\nu = (m, n)$ with m and n natural numbers is called a *normal point* if $mp = nq$ and if the corresponding (square) block Toeplitz matrix $T_{m,n}$ is nonsingular. In the next section, we show that the parameters x_i, z_i of the inversion formula are connected to a right interpolation problem in such a normal point $\nu = (m, n)$ while the parameters u_i, w_i are connected to a left interpolation problem.

3. Connection to a homogeneous interpolation problem

Let $L(z) = \sum_{k=-\infty}^{+\infty} l_k z^k$ be a formal Laurent series. The notation $L(z) = O_+(z^j)$ is used when $l_k = 0, k < j$. Similarly, the notation $L(z) = O_-(z^j)$ means that $l_k = 0, k > j$. Consider the formal Laurent series $T(z) = \sum_{k=-\infty}^{+\infty} t_k z^k$ with coefficients $t_k \in \mathbb{R}^{p\times q}$. In this section, we assume the values of m and n to be fixed.

DEFINITION 2. [right homogeneous interpolation problem] For each natural number α, the set $\mathcal{S}_{R,\alpha}$ is defined as the set of all vector polynomials $q(z) \in \mathbb{R}[z]^{q\times 1}$ satisfying the interpolation condition

$$T(z)q(z) = O_-(z^{-n+\alpha-1}) + O_+(z^m) \tag{3}$$

and the degree condition $\deg q \leq \alpha$.

Note that if $m + n - \alpha \leq 0$ the interpolation condition is trivially satisfied. In this section, we shall prove that for a given α each element of $\mathcal{S}_{R,\alpha}$ can be written in a unique way as a linear polynomial combination of the columns of two matrix polynomials. Therefore, we need the following lemma.

LEMMA 3. *The block Toeplitz matrix $T_{m,n}$ is nonsingular iff all nontrivial solutions of $\mathcal{S}_{R,n}$ have a nonzero highest degree coefficient.*

Proof. The vector q of coefficients of a nontrivial solution $q(z)$ of $\mathcal{S}_{R,n}$ having degree $< n$ should satisfy the set of linear homogeneous equations: $T_{m,n}q = 0$. Hence, such a nontrivial $q(z)$ exists iff $T_{m,n}$ is singular. ■

THEOREM 4. *If the block Toeplitz matrix $T_{m,n}$ is nonsingular, each element $q(z)$ of $\mathcal{S}_{R,\alpha}$ can be written in a unique way as*

$$q(z) = [X(z) \quad Z(z)] \begin{bmatrix} q_x(z) \\ q_z(z) \end{bmatrix}$$

with $\deg q_x \leq \min(\alpha - n - 1, m - 1)$, $\deg q_z \leq \alpha - n$, $Z(z) = -z_0 - z_1 z - \cdots - z_{n-1}z^{n-1} + I_q z^n$, *and* $X(z) = x_0 + x_1 z + \cdots + x_{n-1}z^{n-1}$, *where x_k, z_k are the same as in Theorem 1.*

Proof. The proof is by induction on α. It is clear that the only element of $\mathcal{S}_{R,\alpha}$ for $\alpha < n$ is the zero vector polynomial. For $\alpha = n$, the columns of $Z(z)$ are elements of $\mathcal{S}_{R,\alpha}$. Because its highest degree coefficients are the columns of I_q, hence linearly independent, any element $q(z)$ of $\mathcal{S}_{R,\alpha}$ can be

written in a unique way as $q(z) = Z(z)q_z + q'(z)$ with $\deg q_z \leq 0$ (hence, q_z is constant) and $\deg q' < n$. However, because $q' \in \mathcal{S}_{R,n}$ having degree $\deg q' < n$, $q' = 0$ follows from Lemma 3. This proves the theorem for $\alpha = n$. Suppose the theorem is true for $\alpha < m + n$. Because $T(z)X(z) = O_-(z^{-1}) + I_p + O_+(z^m)$, we can write every element $q(z)$ of $\mathcal{S}_{R,m+n}$ in a unique way as $q(z) = z^m Z(z)q_{z,m} + z^{m-1}X(z)q_{x,m-1} + q'(z)$ by making the highest degree coefficient of q and the coefficient of z^{m-1} in $T(z)q(z)$ zero. In other words, $q'(z) \in \mathcal{S}_{R,\alpha-1}$. Hence, we can write $q(z)$ in a unique way as $q(z) = Z(z)(q_z(z) + z^m q_{z,m}) + X(z)(q_x(z) + z^{m-1}q_{x,m-1})$. This proves the theorem for $\alpha = m + n$.
For $\alpha \geq m + n$, the elements q of $\mathcal{S}_{R,\alpha}$ only have to satisfy the degree condition $\deg q \leq \alpha$. Hence, any element $q(z)$ of $\mathcal{S}_{R,\alpha}$ with $\alpha > m + n$ can be written in a unique way as $q(z) = z^{\alpha-n}Z(z)q_{z,\alpha-n} + q'(z)$ with $q'(z) \in \mathcal{S}_{R,\alpha-1}$. Applying the same reasoning as above this proves the theorem for $\alpha > m + n$. ■
The polynomial matrix $B_R(z) = [X(z) \quad Z(z)]$ is called the *normalized* (right) basis matrix in the normal point $\nu = (m, n)$. It is also possible to construct a basis matrix in points (m, n) which are not normal but we do not need this result in the sequel. It is clear that the polynomial matrix

$$B_R(z)\Gamma_R = [X(z) \quad Z(z)] \begin{bmatrix} \Gamma_x & 0 \\ 0 & \Gamma_z \end{bmatrix}$$

with $\Gamma_R = \text{diag}(\Gamma_x, \Gamma_z)$ and Γ_x and Γ_z nonsingular diagonal (constant) matrices, is also a basis matrix. In Section 7, the diagonal matrix Γ_R will be used to scale each column of $B_R(z)$ in a specific way. In the sequel, we shall work with this scaled basis matrix and denote it with $B_R = [X \quad Z]$. This means that the Γ_x and Γ_z are incorporated in X and Z. If we need the normalized basis matrix, we shall mention it explicitly. Similar to Definition 2, we define the left homogeneous interpolation problem as follows.

DEFINITION 5. [left homogeneous interpolation problem] For each natural number α, the set $\mathcal{S}_{L,\alpha}$ is defined as the set of all vector polynomials $q(z) \in \mathbb{R}[z]^{1\times p}$ satisfying the interpolation condition

$$q(z)T(z) = O_-(z^{-n+\alpha-1}) + O_+(z^m) \tag{4}$$

and the degree condition $\deg q \leq \alpha$.

The next theorem says that all solutions of the left homogeneous interpolation problem can be represented as a linear polynomial combination of the rows of a left basis matrix

$$B_L = \begin{bmatrix} W(z) \\ U(z) \end{bmatrix}$$

with $U(z) = u_0 + u_1 z + \cdots + u_{m-1} z^{m-1}$ and $W(z) = w_0 + w_1 z + \cdots + w_{m-1} z^{m-1} - I_p z^m$, u_k, w_k as in Theorem 1.

THEOREM 6. *If the block Toeplitz matrix $T_{m,n}$ is nonsingular, each element $q(z)$ of $\mathcal{S}_{L,\alpha}$ can be written in a unique way as*

$$q(z) = [q_w(z) \quad q_u(z)] \begin{bmatrix} W(z) \\ U(z) \end{bmatrix}$$

with $\deg q_u \leq \min(\alpha - m - 1, n - 1)$ *and* $\deg q_w \leq \alpha - m$.

Note that when $p = q = 1$, $X(z) = U(z)$ and $Z(z) = W(z)$.

4. Updating procedure

In the previous section, we kept the indices m and n fixed. In this section, we consider all possible normal points $\nu^{(\sigma)} = (m^{(\sigma)}, n^{(\sigma)})$, $\sigma = 0, 1, 2, \ldots$, i.e., such that the (nested) block Toeplitz matrices $T_{m^{(\sigma)}, n^{(\sigma)}}$ based on the data $t_i \in \mathbb{R}^{p \times q}$ are square and nonsingular. Recall that $T_{m,n}$ is square iff $pm = qn$.

If $\nu = (m, n)$ is a normal point and $\nu' = (m', n')$ is one of the subsequent normal points, we shall describe first how we can update the right basis matrix B_R for ν to a right basis matrix for ν' by constructing a polynomial matrix $\hat{B}_R(z)$ such that

$$B'_R(z) = B_R(z) \hat{B}_R(z).$$

From (2) we get for the (scaled) basis matrix

$$T(z)[X(z) \quad Z(z)] = [R_x^-(z) + R_x^+(z) \quad R_z^-(z) + R_z^+(z)]$$

where

$$R_x^-(z) = R_{x,0}^- + R_{x,1}^- z^{-1} + R_{x,2}^- z^{-2} + \cdots, \qquad R_{x,0}^- = \Gamma_x^\nu$$
$$R_x^+(z) = R_{x,1}^+ z^m + R_{x,2}^+ z^{m+1} + R_{x,3}^+ z^{m+2} + \cdots$$

are called the $(-)$-residual series and the $(+)$-residual series for $X(z)$. The series $R_z^-(z)$ and $R_z^+(z)$ are defined by similar relations but with $R_{z,0}^- = 0$.

For $R_v^-(z) = O_-(z^0)$ and $R_v^+(z) = O_+(z^m)$, we define the block Toeplitz matrix $S_{v,r,c}$ as

$$S_{v,r,c} = \begin{bmatrix} R^-_{v,0} & R^-_{v,1} & \cdots & R^-_{v,c-1} \\ 0 & R^-_{v,0} & \cdots & R^-_{v,c-2} \\ \vdots & & \ddots & \vdots \\ 0 & & & \vdots \\ R^+_{v,1} & 0 & \cdots & \vdots \\ R^+_{v,2} & R^+_{v,1} & \ddots & \vdots \\ \vdots & \vdots & & \vdots \\ R^+_{v,r} & R^+_{v,r-1} & \cdots & \end{bmatrix},$$

where the number of block rows is $r + c$ if $c < m$ and $m + r$ if $c \geq m$. The matrix $S_{\nu,\nu'}$ with $\nu = (m, n)$ and $\nu' = (m', n')$ is defined as

$$S_{\nu,\nu'} = [S_{x,\delta_m,\delta} \quad S_{z,\delta_m,\delta_n}] \tag{5}$$

with $\delta_m = m' - m$, $\delta_n = n' - n$ and $\delta = \min(\delta_n, m)$. Note that $S_{\nu,\nu'}$ is square iff $pm' = qn'$. We consider the following system of linear equations

$$S_{\nu,\nu'} \begin{bmatrix} a_0 & c_0 \\ a_1 & c_1 \\ \vdots & \vdots \\ a_{\delta-1} & c_{\delta-1} \\ b_0 & d_0 \\ b_1 & d_1 \\ \vdots & \vdots \\ b_{\delta_n-1} & d_{\delta_n-1} \end{bmatrix} = \begin{bmatrix} I_p & \\ 0 & \\ \vdots & -R_z \\ 0 & \\ 0 & \end{bmatrix} \begin{bmatrix} \Gamma'_x & 0 \\ 0 & \Gamma_z^{-1}\Gamma'_z \end{bmatrix} \tag{6}$$

with $a_j \in \mathbb{R}^{p\times p}$, $b_j \in \mathbb{R}^{q\times p}$, $c_j \in \mathbb{R}^{p\times q}$ and $d_j \in \mathbb{R}^{q\times q}$. In the right hand side, $\Gamma'_R = \mathrm{diag}(\Gamma'_x, \Gamma'_z)$ is the scaling diagonal for $\nu' = (m', n')$ and $\Gamma_R = \mathrm{diag}(\Gamma_x, \Gamma_z)$ is the scaling diagonal for $\nu = (m, n)$. R_z is the block column that would extend S_{z,δ_m,δ_n} one block column to the right. I.e., it has entries $[R^-_{z,\delta_n}, \ldots, R^-_{z,0}, 0, \ldots, 0, R^+_{z,1}, \ldots, R^+_{z,\delta_m-\delta_n}]$. It is proved in the

following theorem that the updating matrix $\hat{B}_R$ can be described with the help of the following polynomial matrices:

$$\begin{aligned} A_R(z) &= a_0 + a_1 z + \cdots a_{\delta-1} z^{\delta-1}, \\ C_R(z) &= c_0 + c_1 z + \cdots c_{\delta-1} z^{\delta-1}, \\ B_R(z) &= b_0 + b_1 z + \cdots b_{\delta_n-1} z^{\delta_n-1}, \\ D_R(z) &= d_0 + d_1 z + \cdots d_{\delta_n-1} z^{\delta_n-1} + \Gamma_z^{-1}\Gamma_z' z^{\delta_n}. \end{aligned}$$

With the notation just introduced, we have

THEOREM 7. *If $B_R(z)$ is a right basis matrix in the normal point $\nu = (m,n)$ then $\nu' = (m',n')$ is also a normal point iff the matrix $S_{\nu,\nu'}$ is nonsingular. Moreover,*

$$B_R'(z) = B_R(z)\hat{B}_R, \qquad \hat{B}_R = \begin{bmatrix} A_R(z) & C_R(z) \\ B_R(z) & D_R(z) \end{bmatrix} \tag{7}$$

is a right basis matrix for ν'.

Proof. Let $\mathcal{S}_{R,\alpha}$ and $\mathcal{S}'_{R,\alpha}$ denote the set $\mathcal{S}_{R,\alpha}$ in the point $\nu = (m,n)$ and $\nu' = (m',n')$ respectively and let $B_R(z) = [X(z) \;\; Z(z)]$. All elements $q(z)$ of $\mathcal{S}'_{R,n'} \subset \mathcal{S}_{R,n'}$ having degree $< n'$ can be written in a unique way as $q(z) = Z(z)q_z(z) + X(z)q_x(z)$ with $\deg q_z < \delta_n$ (equality here would imply that $\deg q = n'$ because the highest degree coefficients of the columns of Z are linearly independent) and $\deg q_x \leq \delta - 1$ where the coefficients of $q_z(z) = q_{z,0} + \cdots + q_{z,\delta_n-1} z^{\delta_n-1}$ and $q_x(z) = q_{x,0} + \cdots + q_{x,\delta-1} z^{\delta-1}$ are solutions of the following system of linear equations

$$S_{\nu,\nu'} \begin{bmatrix} q_{x,0} \\ q_{x,1} \\ \vdots \\ q_{x,\delta-1} \\ q_{z,0} \\ q_{z,1} \\ \vdots \\ q_{z,\delta_n-1} \end{bmatrix} = \begin{bmatrix} 0 \\ \vdots \\ 0 \end{bmatrix}. \tag{8}$$

If the matrix $S_{\nu,\nu'}$ is nonsingular, only the zero vector is a solution of $S_{\nu,\nu'}v = 0$. Hence, from Lemma 3 it follows that ν' is a normal point. On the other hand, if ν' is a normal point, the only solution of (8) should be the zero vector. Hence, $S_{\nu,\nu'}$ should be nonsingular.

If $S_{\nu,\nu'}$ is nonsingular, the reader can easily check that B'_R satisfies the degree and interpolation conditions to be the right basis matrix for the normal point ν'. ■

The previous theorem leads to the updating procedure 8 for computing the right basis matrices $B_R^{(\sigma)}$, $\sigma = 0, 1, \ldots$. To get a *square* block Toeplitz matrix $T_{m,n}$, m and n should satisfy $pm = qn$. The next points (m', n') for which $T_{m',n'}$ is square should satisfy $pm' = qn'$. Thus, $p(m' - m) = q(n' - n) = \lambda$. Hence, the smallest possible increase in $\nu = (m, n)$ is obtained by making λ as small as possible, i.e. by taking λ equal to the least common multiple of p and q. Thus ν can only increase with minimal steps $\Delta = (\lambda/p, \lambda/q)$.

ALGORITHM 8 (UPDATING PROCEDURE)

$\nu^{(0)} \leftarrow (0, 0)$
$B_R^{(0)} \leftarrow [0 \;\; \Gamma_z^{(0)}]$ *($\Gamma_z^{(0)}$ is arbitrary nonsingular diagonal)*
λ is the least common multiple of p and q
$\Delta \leftarrow (\lambda/p, \lambda/q)$; $\sigma \leftarrow 0$
repeat
    $\nu' \leftarrow \nu^{(\sigma)} + \Delta \qquad \{\nu' = (m', n')\}$
    while $S_{\nu^{(\sigma)},\nu'}$ *is singular* **do**
        $\nu' \leftarrow \nu' + \Delta$
    endwhile
    $\nu^{(\sigma+1)} \leftarrow \nu'$
    compute $\hat{B}^{(\sigma)}$, i.e., the polynomial matrices $A_R(z), B_R(z), C_R(z)$, $D_R(z)$ by solving (6) with $\nu = \nu^{(\sigma)}$ and $\nu' = \nu^{(\sigma+1)}$
    $B_R^{(\sigma+1)} \leftarrow B_R^{(\sigma)} \hat{B}_R^{(\sigma)}$
    $\sigma \leftarrow \sigma + 1$
until $m' = M$

This algorithm allows to update $X(z), Z(z)$, part of the inversion parameters, for a sequence of nested block Toeplitz matrices $T_{m^{(\sigma)},n^{(\sigma)}}$ until the desired matrix $T_{M,N}$ is reached. The remaining inversion parameters can be interpreted as building blocks of a basis matrix for the left homogeneous interpolation problem. Similar to the updating procedure for the right basis matrix, we can give an updating procedure for the left basis matrix. Note that the normal points of the left and the right problem are just the same because they are defined by looking at the singularity of the block Toeplitz matrices $T_{m,n}$. Let $\nu = (m, n)$ and $\nu' = (m', n')$ be normal points as above with (scaled) left basis matrices B_L and B'_L respectively. The scaling diagonals are $\Gamma_L = \mathrm{diag}(\Gamma_w, \Gamma_u)$ and $\Gamma'_L = \mathrm{diag}(\Gamma'_w, \Gamma'_u)$ respectively.

We introduce the following notation:

$$B_L(z)T(z) = \begin{bmatrix} W(z) \\ U(z) \end{bmatrix} T(z) = \begin{bmatrix} R_w^-(z) + R_w^+(z) \\ R_u^-(z) + R_u^+(z) \end{bmatrix}$$

with $R_u^-(z) = z^{m-n}(R_{u,0}^- + R_{u,1}^- z^{-1} + R_{u,2}^- z^{-2} + \cdots)$ with $R_{u,0}^- = \Gamma_u$ and $R_u^+(z) = R_{u,1}^+ z^m + R_{u,2}^+ z^{m+1} + R_{u,3}^+ z^{m+2} + \cdots$. Similarly for $R_w^-(z)$ and $R_w^+(z)$ with $R_{w,0}^- = 0$.

For given $\nu = (m,n)$ and $\nu' = (m',n')$ and series $R_v^-(z) = z^{m-n}(R_{v,0}^- + R_{v,1}^- z^{-1} + R_{v,2}^- z^{-2} + \cdots)$ and $R_v^+(z) = R_{v,1}^+ z^m + R_{v,2}^+ z^{m+1} + R_{v,3}^+ z^{m+2} + \cdots$, we define the block Hankel matrix $H_{v,\delta_m,\delta}$ as

$$H_{v,\delta_m,\delta} = \begin{bmatrix} R_{v,\delta_m}^+ & R_{v,\delta_m-1}^+ & \cdots & \cdots & \\ R_{v,\delta_m-1}^+ & \cdots & & & \vdots \\ \vdots & & & \cdots & R_{v,\delta+\delta_n-\delta_m-2}^- \\ & \cdots & \cdots & R_{v,\delta+\delta_n-\delta_m-2}^- & R_{v,\delta+\delta_n-\delta_m-1}^- \end{bmatrix}$$

with $\delta_m = m' - m$ and $\delta_n = n' - n$. The number of block rows is δ while the number of block columns is n' if $\delta \geq n$ and $\delta_n + \delta$ if $\delta < n$. The elements with a negative index corresponding to $(-)$-series are replaced by zeros. Similarly for the elements with a nonpositive index corresponding to $(+)$-series. The matrix $H_{\nu,\nu'}$ is defined as

$$H_{\nu,\nu'} = \begin{bmatrix} H_{w,\delta_m,\delta_m} \\ H_{u,\delta_m,\delta} \end{bmatrix}$$

with $\delta = \min(\delta_m, n)$. Note that $H_{\nu,\nu'}$ is square iff $pm' = qn'$. We consider the following system of linear equations

$$\begin{bmatrix} a_0 & \cdots & a_{\delta_m-1} & b_0 & \cdots & b_{\delta-1} \\ c_0 & \cdots & c_{\delta_m-1} & d_0 & \cdots & d_{\delta-1} \end{bmatrix} H_{\nu,\nu'} = \begin{bmatrix} \Gamma_w' \Gamma_w^{-1} & 0 \\ 0 & \Gamma_u' \end{bmatrix} \begin{bmatrix} -R_w \\ 0 \cdots 0 \;\; I_q \end{bmatrix} \tag{9}$$

with $a_j \in \mathbb{R}^{p\times p}$, $b_j \in \mathbb{R}^{p\times q}$, $c_j \in \mathbb{R}^{q\times p}$ and $d_j \in \mathbb{R}^{q\times q}$ where R_w is the block row that would extend H_{w,δ_m,δ_m} with one block row at the bottom, i.e.,

$$R_w = [0, \ldots, 0, R_{w,1}^-, \ldots, R_{w,\delta_n}^-].$$

We define the polynomial matrices A_L, B_L, C_L, D_L as

$$\begin{aligned} A_L(z) &= a_0 + a_1 z + \cdots a_{\delta_m-1} z^{\delta_m-1} + \Gamma_w' \Gamma_w^{-1} z^{\delta_m} \\ C_L(z) &= c_0 + c_1 z + \cdots c_{\delta_m-1} z^{\delta_m-1} \\ B_L(z) &= b_0 + b_1 z + \cdots b_{\delta-1} z^{\delta-1} \\ D_L(z) &= d_0 + d_1 z + \cdots d_{\delta-1} z^{\delta-1}. \end{aligned}$$

THEOREM 9. *If $B_L(z)$ is a left basis matrix in the normal point $\nu = (m,n)$ then $\nu' = (m',n')$ is also a normal point iff the matrix $H_{\nu,\nu'}$ is nonsingular. Moreover,*

$$B'_L(z) = \hat{B}_L(z)B_L(z), \qquad \hat{B}_L(z) = \begin{bmatrix} A_L(z) & B_L(z) \\ C_L(z) & D_L(z) \end{bmatrix} \tag{10}$$

is a left basis matrix $B'_L(z)$ for the normal point ν'.

Proof. The proof is similar to the proof of Theorem 7. ■

The previous theorem leads to an updating procedure for left basis matrices similar to the one for right basis matrices.

5. Duality

To compute the parameters X, Z, U, W of the inversion formula for a block Toeplitz matrix, we will design Algorithm 16. Because the error analysis of this algorithm is similar to the analysis for generalized Sylvester matrices in [4, 2, 3], we shall only mention the main results in Sections 8-10. For more details and proofs, we refer the reader to our report [28]. If possible also the link is given to the corresponding result in [4, 2, 3] where we use the same indexing of the θ entities as in [4]. In the detailed error analysis of [4] and [28] some duality results are needed, but these are interesting enough in their own right to be mentioned separately in this section.

We fix the values m and n. If the interpolation condition (3) is not satisfied exactly, but

$$T(z)[X(z) \;\; Z(z)] = [\delta R_x(z) + R_x^+(z) + R_x^-(z) \;\; \delta R_z(z) + R_z^+(z) + R_z^-(z)], \tag{11}$$

with $\delta R_x(z) = O_+(z^0)$ and $\delta R_z(z) = O_+(z^0)$, then we call $\delta R_x(z)$ and $\delta R_z(z)$ the residual errors and $B_R = [X \;\; Z]$ a right *numerical* basis matrix (NBM). The result of the error analysis as given by Theorem 17 shows that under certain conditions the residual errors can be kept small when measured by certain norms to be defined in Section 6. We can write (11) more compactly as

$$T(z)B_R(z) = \delta R_R(z) + R_R^+(z) + R_R^-(z) \quad \text{with} \quad \begin{bmatrix} \delta R_R \\ R_R^+ \\ R_R^- \end{bmatrix} = \begin{bmatrix} \delta R_x & \delta R_z \\ R_x^+ & R_z^+ \\ R_x^- & R_z^- \end{bmatrix}. \tag{12}$$

In the error analysis, a duality result is needed, which can be written in terms of *extended* basis matrices. The extended basis matrices have the

advantage that they are square. Thus their determinant is defined and they are potentially invertible.

DEFINITION 10. [extended basis matrix] A right extended basis matrix (EBM) is defined as

$$\overline{B}_R = \begin{bmatrix} R_x^+ & R_z^+ \\ X & Z \end{bmatrix} = \begin{bmatrix} R_R^+ \\ B_R \end{bmatrix}.$$

Similarly an extended left basis matrix is defined as

$$\overline{B}_L = \begin{bmatrix} W & -R_w^+ \\ U & -R_u^+ \end{bmatrix} = [B_L \quad -R_L^+].$$

If B_R or B_L are numerical basis matrices then $\overline{B}_R$ and $\overline{B}_L$ are called extended numerical basis matrices (ENBM).

Similar definitions could be given for the $(-)$-series but we do not need them in the sequel. Note that in [4], the basis matrices are already "extended", i.e. are square in a natural way. Here, we have to introduce the $(+)$-series in a rather artificial way to make the resulting basis matrix square. Before we give the next lemma, we need the definition of the projection operator.

DEFINITION 11. [projection operator] For a formal Laurent series $T(z) = \sum_{k=-\infty}^{+\infty} t_k z^k$, we define the projection operator $\Pi_{m:n}$ as

$$\Pi_{m:n} T(z) = \sum_{k=m}^{n} t_k z^k.$$

LEMMA 12. *If $\overline{B}_R(z)$ is a right ENBM, then*

$$(-1)^p \det \overline{B}_R(z) = \gamma z^{pm} + \theta_I(z)$$

with

$$\begin{aligned} \theta_I(z) &= \Pi_{pm:+\infty} \left(\det \begin{bmatrix} \delta R_x(z) & \delta R_z(z) \\ X(z) & Z(z) \end{bmatrix} \right) \qquad \textit{and} \\ \gamma &= \det \Gamma_R. \end{aligned} \tag{13}$$

Proof. The proof can be found in [28] Lemma 6.1. See also Lemma 1 in [4]. ■
The previous lemma states that for a right EBM the determinant is a monomial γz^{pm}. The next duality result is used in the proofs of the error analysis in [28].

THEOREM 13 (DUALITY) *If $\overline{B}_L(z)$ and $\overline{B}_R(z)$ are left and right ENBM respectively, then*

$$\overline{B}_L(z)\overline{B}_R(z) = z^m \Gamma_L \Gamma_R + \theta_{II}(z),$$

where

$$\theta_{II}(z) = \Pi_{m:+\infty} \left(\delta R_L(z) B_R(z) - B_L(z)\delta R_R(z)\right) \tag{14}$$

Proof. The proof can be found in [28] Theorem 6.2 and is very similar to the proof of Theorem 4 in [4]. See also Theorem 1 in [2]. ■
The next two corollaries define the quantities θ_{III} and θ_{IV} which are used in the formulation of Theorem 18.

COROLLARY 14. *If $\overline{B}_R(z)$ is a right normalized and $\overline{B}_L(z)$ a left normalized ENBM for the normal point (m, n) of $T(z)$, then*

$$\overline{B}_R(z)\overline{B}_L(z) = z^m I_{p+q} + \theta_{III}(z)$$

with

$$\theta_{III}(z) = \overline{B}_R(z)\Gamma_L^{-1}\theta_{II}(z)\Gamma_R^{-1}\overline{B}_R^{-1}(z).$$

Proof. See the proof of Corollary 6.1 of [28] and the similar Corollaries 5 of [4] and 2 of [2]. ■
When in the previous corollary, $\theta_{III} = 0$, i.e. $\overline{B}_L$ and $\overline{B}_R$ are normalized EBMs, then they are inverses of each other except for the factor z^m.

COROLLARY 15. *The residual series $R_R^-(z)$ for a normalized ENBM $\overline{B}_R(z)$ satisfies*

$$R_R^-(z)\overline{B}_L(z) = \overline{T}(z)z^m + \theta_{IV}(z)$$

with $\overline{T}(z) = [-I_p \;\; T(z)]$ and $\theta_{IV}(z) = \overline{T}(z)\theta_{III}(z) - \delta R_R(z)\overline{B}_L(z)$.

Proof. See the proof of Corollary 6.2 of [28] and the similar Corollaries 6 of [4] and 3 of [2]. ■

6. Norms

Before we give the updating procedure for the extended basis matrix, we give the definitions of the norms that will be used in the error analysis of Section 8. We will assume that all the series considered have only a finite number of nonzero coefficients. This is no limitation for practical applications where we can assume that all coefficients of $T(z)$ not appearing in the block Toeplitz matrix $T_{M,N}$ are zero. So, we can define a norm for polynomials, polynomial vectors and matrices as well as for series. The set of polynomials $\mathbb{R}[z]$ is considered as a subset of $\mathbb{R}(z)$, i.e. the set of all formal Laurent series (with only a finite number of nonzero coefficients). The norm of $a(z) = \sum_{i=-\infty}^{+\infty} a_i z^i \in \mathbb{R}(z)$ is defined as

$$||a(z)|| = \sum |a_i|$$

where the sum is taken over all nonzero coefficients. The norm of $A(z) \in \mathbb{R}(z)^{p\times q}$ is defined as

$$||A(z)|| = \max_{j=1,\ldots,q} \sum_{i=1}^{p} ||a_{i,j}(z)||.$$

It is easy to prove that for two elements A_1 and A_2 of $\mathbb{R}(z)^{p\times q}$: $||A_1 A_2|| \leq ||A_1||\,||A_2||$, $||A_1|| \leq p||A_1^T||$ and $||A_1^T|| \leq q||A_1||$. Note that in [4] different norms are used for polynomials and series. This makes the proofs harder to read but the advantage is that no assumption has to be made concerning the number of nonzero coefficients of a series.

7. Updating procedure for the extended basis matrix

We shall now give an updating procedure for the ENBMs $\overline{B}_R^{(\sigma)}$ and $\overline{B}_L^{(\sigma)}$ for the successive normal points $\nu^{(\sigma)} = (m^{(\sigma)}, n^{(\sigma)})$, $\sigma = 0, 1, 2, \ldots$. The updating for the right ENBM will be typically of the form (see Theorem 7)

$$\overline{B}_R^{(\sigma+1)} = \overline{B}_R^{(\sigma)} \hat{B}_R^{(\sigma)}.$$

This update however is only valid if $\delta_n^{(\sigma)} = n^{(\sigma+1)} - n^{(\sigma)} \leq m^{(\sigma)}$. If $\delta_n^{(\sigma)} > m^{(\sigma)}$, also the $R_z^{(\sigma)-}(z)$ series is needed to update the $R_R^{(\sigma)+}(z)$ series. In this case the dimension of the matrix $S^{(\sigma)} = S_{\nu^{(\sigma)},\nu^{(\sigma+1)}}$ is equal to the dimension of the matrix $T_{m^{(\sigma+1)},n^{(\sigma+1)}}$. Hence, we can just use $T_{m^{(\sigma+1)},n^{(\sigma+1)}}$ to compute an extended basis matrix. A similar reasoning is

true for updating the left ENBM. Thus if $n^{(\sigma+1)} - n^{(\sigma)} > m^{(\sigma)}$ or $m^{(\sigma+1)} - m^{(\sigma)} > n^{(\sigma)}$, we forget the previous computations and start all over just like in the first step where the basis matrices for $\sigma = 1$ are computed. This leads to the updating procedure 16. The formal Laurent series $T(z)$ is scaled such that $\max\{||T(z)||, ||T(z)^T||\} \leq 1$. This can be done by first scaling the columns of $T(z)$ such that they get a norm equal to one and then scaling the rows of $T(z)$ having a norm greater than one in the same way. The downscaling of the rows will not increase the norms of the columns. The diagonal matrices $\Gamma_L^{(\sigma)}$ and $\Gamma_R^{(\sigma)}$ are chosen such that $||\overline{B}_R^{(\sigma)}(z)|| = ||\overline{B}_L^{(\sigma)}(z)^T|| = 1$. The error analysis will show that the quantity

$$\kappa^{(\sigma)} = \det(\Gamma_L^{(\sigma)}\Gamma_R^{(\sigma)})^{-1} \tag{15}$$

gives a measure for the stability of the point $\nu^{(\sigma)} = (m^{(\sigma)}, n^{(\sigma)})$ (see Theorem 19). This quantity is compared to a user-supplied tolerance value τ. If $\kappa^{(\sigma)} \leq \tau$, we say that the point $\nu^{(\sigma)}$ is stable. The final value for m' in the algorithm is M. The variable *choice* indicates at each step whether the solution was accepted or not, based on the comparison between $\kappa^{(\sigma)}$ and τ. It also indicates whether the extended basis matrices are computed using (part of) the given (square) block Toeplitz matrix $T_{M,N}$ ($separated = false$) or using the updating formulas ($separated = true$). Thus *choice* has the following meaning

$$choice = \begin{cases} -1 & \text{not separated and } T_{m',n'} \text{ not stable} \\ 0 & \text{separated and } S^{(\sigma)} \text{ not stable} \\ 1 & \text{separated and } S^{(\sigma)} \text{ stable} \\ 2 & \text{not separated and } T_{m',n'} \text{ stable} \end{cases}$$

Thus if $choice \leq 0$, the solution is rejected for the updating scheme (look-ahead step) and if $choice > 0$, the solution is accepted for updating. The value of *choice* will be plotted in Section 12 when showing some experimental results. To construct the matrix $S^{(\sigma)}$ (similarly for $H^{(\sigma)} = H_{\nu^{(\sigma)},\nu^{(\sigma+1)}}$) the entries needed from the $(+)$-series can be directly taken from the ENBM $\overline{B}_R^{(\sigma)}$. As part of this ENBM, the $(+)$-series are updated completely in each step. Hence, this part of the algorithm could be called Schur-type. On the other hand during each step only the entries of the $(-)$-series needed in the matrix $S^{(\sigma)}$ are computed by inner products using (11). This is of Levinson-type. It is clear that several combinations are possible. However, to be able to follow a similar error analysis as in [4, 2, 3], we have chosen for a Schur-type update of the $(+)$-series and a Levinson-type computation to derive part of the $(-)$-series in each step.

ALGORITHM 16 (UPDATING PROCEDURE (EXTENDED BASIS MATRIX))

UPDATING$(T(z), M, \tau)$
$\lambda \leftarrow$ *least common multiple* (p, q)
$\Delta \leftarrow (\lambda/p, \lambda/q)$
$\nu' \leftarrow (0,0)$; $\nu^{(0)} \leftarrow (0,0)$; $\sigma \leftarrow 0$
repeat
  $\nu' \leftarrow \nu' + \Delta \qquad \{\nu' = (m', n')\}$
  separated $\leftarrow$ $(n' - n^{(\sigma)} \le m^{(\sigma)})$ *and* $(m' - m^{(\sigma)} \le n^{(\sigma)})$
  if *not separated* **then**
    if $T_{m',n'}$ *is numerically nonsingular* **then**
      compute $\overline{B}_R^{(1)}(z)$ *and* $\overline{B}_L^{(1)}(z)$ *(using Def. 10, (11) and Th. 1)*
      stable $\leftarrow \kappa^{(1)} \le \tau$
      if *stable* **then**
        $\nu^1 \leftarrow \nu'$
        $\sigma \leftarrow 1$
        choice $\leftarrow 2$
      else
        choice $\leftarrow -1$
      endif
    endif
  else if $S_{\nu^{(\sigma)},\nu'}$ *is numerically nonsingular* **then**
      $\nu^{(\sigma+1)} \leftarrow \nu'$
      compute $A_R(z), B_R(z), C_R(z), D_R(z)$
          (Gaussian elimination on $S^{(\sigma)T}$ *of (6))*
      $\overline{B}_R^{(\sigma+1)}(z) \leftarrow \overline{B}_R^{(\sigma)} \hat{B}_R^{(\sigma)}(z)$
      compute $A_L(z), B_L(z), C_L(z), D_L(z)$
          (Gaussian elimination on $H^{(\sigma)}$ *of (9))*
      $\overline{B}_L^{(\sigma+1)}(z) \leftarrow \hat{B}_L^{(\sigma)}(z) \overline{B}_L^{(\sigma)}$
      stable $\leftarrow \kappa^{(\sigma+1)} \le \tau$
      if *stable* **then**
        $\sigma \leftarrow \sigma + 1$
        choice $\leftarrow 1$
      else
        choice $\leftarrow 0$
      endif
  endif
until $m' = M$
if *stable* **then**
  return $(\overline{B}_R^{(\sigma)}(z), \overline{B}_L^{(\sigma)}(z), \kappa^{(\sigma)})$
else

return $(\overline{B}_R^{(\sigma+1)}(z), \overline{B}_L^{(\sigma+1)}(z), \kappa^{(\sigma+1)})$
endif

If the order of the block Toeplitz matrices $T_{m',n'}$ and of the matrices $S^{(\sigma)}$ and $H^{(\sigma)}$ are small during the algorithm, this algorithm requires $O((pM)^2)$ floating point operations. Hence, this algorithm is an order of magnitude faster than using the classical algorithms to compute the final basis matrix from the block Toeplitz system of equations of order pM without taking into account the special structure. Hence, Algorithm 16 can be called fast. In the literature, there have appeared also so-called superfast algorithms. However, the stability of these algorithms is still an open question.

8. Error analysis

Because the error analysis is similar to the one presented in [4, 2, 3], we only give the final result here and refer the interested reader to [28] for the details. Algorithm 16 computes the ENBMs $\overline{B}_L^{(\sigma)}$ and $\overline{B}_R^{(\sigma)}$ for $\sigma = 0, 1, \ldots, \Sigma$ such that the points $(m^{(\sigma)}, n^{(\sigma)})$ are stable (except possibly for the last point $(m^{(\Sigma)}, n^{(\Sigma)}) = (M, N)$), i.e., such that the corresponding quantity $\kappa^{(\sigma)}$ is smaller than a threshold value τ. The error analysis gives us an upper bound for $\|\delta R_R^{(\sigma+1)}(z)\|$, the norm of the residual error.

THEOREM 17. *Let μ denote the unit roundoff or machine precision. Let $\delta_n^{(\sigma)} = n^{(\sigma)} - n^{(\sigma-1)}$ and $\delta_m^{(\sigma)} = m^{(\sigma)} - m^{(\sigma-1)}$ for $\sigma = 1, 2 \ldots, \Sigma$. The residual errors $\delta R_R^{(j)}(z)$ and $\delta R_L^{(j)}(z)$ are defined by (12) and (11) for the point $(m^{(j)}, n^{(j)})$. The quantity $\kappa^{(\sigma)}$ is defined by (15). If μ, $\|\delta R_R^{(j)}(z)\|$ and $\|\delta R_L^{(j)}(z)\|$ are so small that*

$$(N + p + q)\mu \leq 0.01$$

and

$$\begin{aligned}\kappa^{(j)} \Big\{ &\Big((p+q)\|\delta R_R^{(j)}(z)\| + \|\delta R_L^{(j)}(z)\|\Big) \\ &+ \Big((p+q)1.01(\delta_n^{(j)} + p + q)\mu \\ &+(p+q)8p^3(\delta_m^{(j)} + \delta_n^{(j)})^3\rho_j\mu + (p+q)f_j(\mu)\Big)\Big\} \leq \frac{1}{2}, \quad j \leq \sigma,\end{aligned}$$

with ρ_j the growth factor of the Gaussian elimination and $f_j(\mu)$ the $O(\mu^2)$ term appearing in the error analysis of the Gaussian elimination on $\left(S^{(j)}\right)^T$

(see Lemma 9.4 of [28]), then

$$\|\delta R_R^{(\sigma+1)}(z)\| \leq F_\sigma + 2(p+q)\sum_{j=0}^{\sigma-1}\kappa^{(j+1)}F_j$$

with

$$\begin{aligned} F_j &= 4\kappa^{(j)}(p+q)\mu \\ &\quad \left\{(n^{(j)}+q) + 4p^3(\delta_n^{(j)}+\delta_m^{(j)})^3\rho_j + (\delta_n^{(j)}+p+q)\right\}. \end{aligned} \tag{16}$$

Proof. The proof of this theorem is given in [28] Theorem 9.13. See also Theorem 22 of [4] and Theorem 12 of [3]. ■
This theorem shows that if $\|\delta R_R^{(j)}(z)\|$ and $\|\delta R_L^{(j)}(z)\|$ are small for $j \leq \sigma$ also $\|\delta R_R^{(\sigma+1)}(z)\|$ will be small if $\kappa^{(\sigma)}$ is not too large. An analogous theorem holds for $\delta R_L^{(\sigma+1)}(z)$.

9. The inverse of a block Toeplitz matrix

In this section, we give formulas for the inverse of a block Toeplitz matrix based on *numerical* basis matrices. These results will be used in the next section to get an upper bound for the norm of the error between the exact and the computed right basis matrix. Let $L(z) = \sum_{k=-\infty}^{+\infty} l_k z^k$ be a formal Laurent series. Then $[L^{f(i,j)}]$ denotes a matrix whose element at position (i,j) is $l_{f(i,j)}$.

THEOREM 18. *If $B_R(z)$ is a right and $B_L(z)$ a left* normalized *NBM for the normal point (m,n) of $T(z)$, the inverse of the block Toeplitz matrix $T_{m,n}$ satisfies*

$$\begin{aligned} T_{m,n}^{-1}(I+\theta_{VIII}) &= L_{(n,\eta)}(z_0,z_1,\ldots,z_{n-1})U_{(\eta,m)}(0,u_{m-1},\ldots,u_1) \qquad (17)\\ &\quad + L_{(n,\eta)}(x_0,x_1,\ldots,x_{n-1})U_{(\eta,m)}(I_p,-w_{m-1},\ldots,-w_1) \\ &= M_{m,n} \qquad (18) \end{aligned}$$

with

$$\begin{aligned} \theta_{VIII} &= \left[\theta_{IV,L}^{(m+i-j)}\right] - \left[T^{(-n+i-j)}\right]\left[\theta_{III,L,L}^{(m+n+i-j)}\right] \\ &\quad + \left[\delta R_x^{(i-j)}\right]\left[W^{(m+i-j)}\right] + \left[\delta R_z^{(i-j)}\right]\left[U^{(m+i-j)}\right]. \end{aligned}$$

Here, $\theta_{IV,L}(z)$ denotes the left block of $\theta_{IV}(z)$ (defined in Corollary 15), i.e. the first p columns. Similarly, $\theta_{III,L,L}(z)$ denotes the lower, left block of $\theta_{III}(z)$ (defined in Corollary 14), i.e. the last q rows of the first p columns of $\theta_{III}(z)$.

Proof. See Theorem 10.1 of [28], Theorem 34 of [4] and Theorem 4 of [2]. ■

Note that this theorem shows that $M_{m,n}$ is the inverse of $T_{m,n}$ if all errors are zero, i.e. if $B_R(z)$ and $B_L(z)$ are *exact* basis matrices.

10. Stability

In this section, we investigate how accurately the scaled right basis matrix is computed. In the sequel, we will drop the index (σ), i.e., $B_R(z) = B_R^{(\sigma)}(z)$, $\kappa = \kappa^{(\sigma)}$, ... In Section 8, we have given an upper bound for $||\delta R_R(z)||$. A similar one holds for $||\delta R_L(z)||$. We will use this result to give an upper bound for $||\delta B_R(z)||$ with $\delta B_R(z) = B_R(z) - B_{R,E}(z)$, where $B_{R,E}(z)$ is the scaled *exact* BM with the same scaling as $B_R(z)$. Because

$$T(z)\delta B_R(z) = \Pi_{0:m-1}\left(\delta R_R(z)\right) + O_-(z^{-1}) + O_+(z^m),$$

we get that

$$||\delta B_R(z)|| \leq ||T_{m,n}^{-1}||_1 ||\delta R_R(z)||.$$

THEOREM 19. *If μ is so small and $||\delta R_R(z)||$ and $||\delta R_L(z)||$ are not too large such that the conditions of Theorem 17 are satisfied and*

$$(p+q)!||\delta R_R(z)||^p \leq \gamma/2 \tag{19}$$

with γ defined by (13) and

$$2\kappa(p+q)||\delta R_R(z)|| + \frac{4\kappa(p+q)!}{\gamma}\left\{(p+q)||\delta R_R(z)|| + ||\delta R_L(z)||\right\} \leq 1/2, \tag{20}$$

then

$$||\delta B_R(z)|| \leq 2\kappa(p+q)\left\{F_\sigma + 2(p+q)\sum_{j=0}^{\sigma-1}\kappa^{(j+1)}F_j\right\}$$

where F_j is defined by (16).

Proof. See Theorem 11.1 of [28], Theorem 40 of [4] and Theorem 15 of [3]. ■

Hence, if we allow in Algorithm 16 only those points $\nu^{(\sigma)} = (m^{(\sigma)}, n^{(\sigma)})$ where the $\kappa^{(\sigma)}$ value is smaller than a tolerance value τ (stable points), the conditions of Theorem 19 will be satisfied. Hence, the right basis matrices will be accurately computed. It turns out that under the conditions of the previous theorem, the 1-norm condition number of $T_{m^{(\sigma)},n^{(\sigma)}}$ is bounded by $2\kappa^{(\sigma)}(p+q)$. So, algorithm 16 is a weakly stable algorithm to compute the right basis matrices in the stable points.

11. The solution of a block Toeplitz system

In this section, we investigate the final error by using the approximation $M = M_{m,n}$ given by (18) of the inverse of the block Toeplitz matrix $T = T_{m,n}$ to solve a block Toeplitz system of linear equations

$$TV = Y, \qquad T \in \mathbb{R}^{pm \times qn}, V \in \mathbb{R}^{qn \times 1}, Y \in \mathbb{R}^{pm \times 1}$$

where T and Y are given and V is the unknown vector.

THEOREM 20. *If the conditions of Theorem 19 are satisfied, then the computed solution V of the equation*

$$TV = Y$$

satisfies

$$V = V_E + \theta_{XI}$$

with

$$\|\theta_{XI}\|_1 \leq (p+q)\left[\kappa + 1.01\mu(2m+2n+3p+q+2) + O(\mu^2)\right]\|Y\|_1$$

and

$$\|Y\|_1 \leq 2\kappa(p+q)\|V_E\|_1,$$

with

$$V_E = T^{-1}Y.$$

Proof. See Theorem 12.2 of [28]. ■

This shows that executing Algorithm 16 and multiplying the approximating inverse M by the right-hand side Y gives a weakly stable algorithm to solve the block Toeplitz system of linear equations.

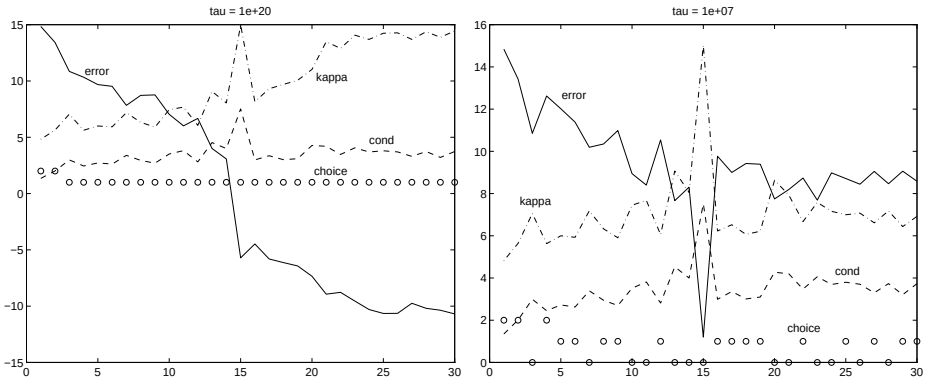


Figure 1: *error*, κ, *choice* and $\mathrm{cond}(T_{m,n})$ for $\tau = 10^{20}$ and $\tau = 10^{7}$

12. Experimental results

In this section, we show how accurately Algorithm 16 together with the inversion formula computes the solution of each block-Toeplitz set of linear equations

$$T_{m,n} V_{m,n} = Y_{m,n}$$

for each $T_{m,n}$ appearing explicitly or implicitly in Algorithm 16. The elements of the block Toeplitz matrix $T_{M,N}$ were taken as random numbers uniformly distributed in the interval $(0,1)$ except for one element in the first row making at least one leading principal submatrix ill-conditioned. The corresponding formal Laurent series $T(z)$ was scaled as indicated in Section 7 such that $\max\{\|T(z)\|, \|T(z)^T\|\} \leq 1$ leading to the final elements of $T_{M,N}$. The right-hand side $Y_{m,n}$ of the system of linear equations $T_{m,n} V_{m,n} = Y_{m,n}$ is computed by setting $V_{m,n} = [1, 1, \ldots, 1]^T$. All the computations were done in MATLAB Version 4.2a and 4.2c on a DEC 5000/240 workstation. The machine precision μ is of the order 10^{-16}.

EXAMPLE 21. In Figure 1, we show the influence of the threshold parameter τ. We take $M = 60$, $N = 90$, $p = 3$ and $q = 2$. If we set $V_{E,m,n} = [1, 1, \ldots, 1]^T$, the plot *error* gives for each step $-\log_{10}(\|V_{m,n} - V_{E,m,n}\|_1 / \|V_{m,n}\|_1)$ where $V_{m,n}$ is computed by using the inversion formula. Also $\log_{10}(\kappa)$ and *choice* appearing in Algorithm 16 are plotted as well as the $\log_{10}$ of the (2-norm) condition number of $T_{m,n}$. Note the big loss of accuracy corresponding to the 15th block leading principal submatrix which is ill-conditioned. When the threshold parameter τ is lowered, we avoid these bad intermediate results. In this case, the plots of κ and the condition

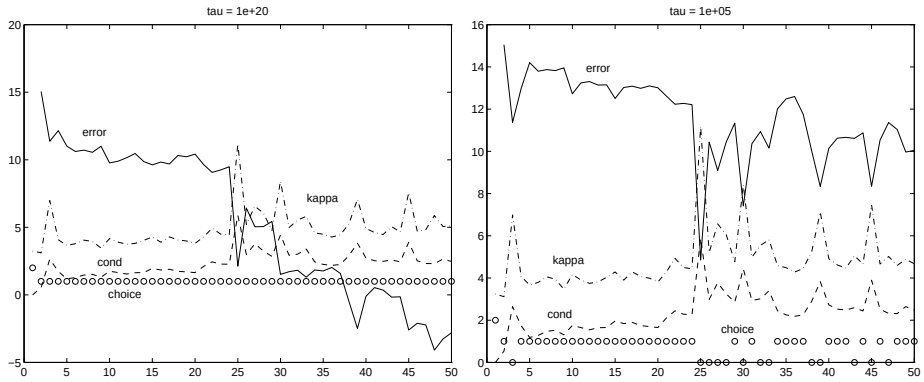


Figure 2: *error*, κ, *choice* and $\mathrm{cond}(T_{m,n})$ for $\tau = 10^{20}$ and $\tau = 10^5$

number have a similar shape showing that κ gives a good estimate of the relative increase or decrease of the condition number of $T_{m,n}$.

EXAMPLE 22. Figure 2 shows the influence of the threshold parameter τ for an example with $M = N = 50$, $p = q = 1$ where the 25th leading principal submatrix is ill-conditioned. Note that in this scalar case, $X(z) = U(z)$ and $Z(z) = W(z)$.

13. Numerical results using another look-ahead criterion

The problem with the look-ahead scheme used in Algorithm 16 is that the user has to choose the threshold value τ. The choice of this τ determines where and how big the look-ahead steps are taken. There is no limit on the size of these look-ahead steps. In this section, we want to develop a more sophisticated look-ahead criterion inspired by the one given by Chan and Hansen in [7, 8]. The user gives $h_{\max}$ which is the maximal length of a look-ahead step. During the algorithm κ is compared to a reference value $\bar{\kappa}$. This reference value is initialized as the minimal κ-value of the first $h_{\max}$ steps. The point where this minimum occurs is the first look-ahead step. From then on, we look for the first step with $\kappa < 2\bar{\kappa}$ within the next $h_{\max}$ steps. The algorithm runs more efficiently as long as the steps taken are smaller. Normally, one expects steps of length 1 except if the κ-values are not comparable to $\bar{\kappa}$ within the next $h_{\max}$ steps. If $\kappa \geq 2\bar{\kappa}$ for all these

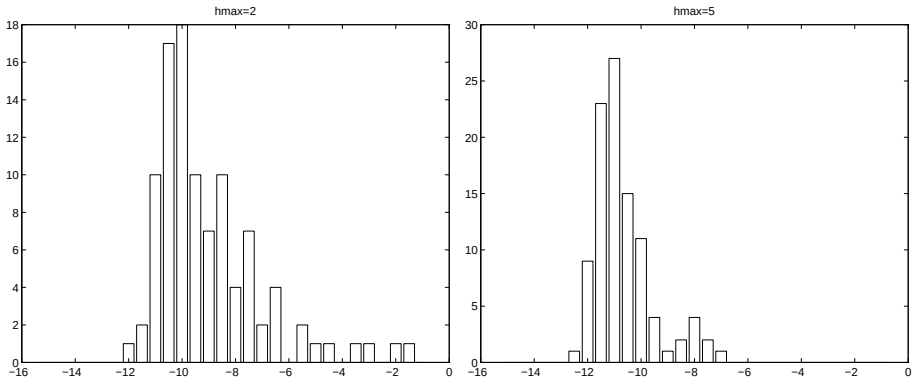


Figure 3: Histogram of $\log_{10}$(relative error)

steps, we take the step with the smallest κ as the next look-ahead step and set $\bar{\kappa}$ equal to this smallest κ value.

EXAMPLE 23. Figure 3 shows the effect of taking another value for $h_{\max}$. We have taken 100 scalar Toeplitz matrices of size 64×64 to compare with the results of Example 1 of [10]. The Toeplitz matrices have at least one principal submatrix which is very ill-conditioned. Figure 3 shows a histogram for the relative error of the solution for $h_{\max} = 2$ and $h_{\max} = 5$. We see that our results ar spread over a large range, for $h_{\max} = 2$ between -13 and -1, for $h_{\max} = 5$ between -13 and -6 compared to the results of [10] which are between -14 and -10. The range will become much smaller by refining our look-ahead criterion in Section 15. Also, compared to [10], the smallest relative errors obtained are still a factor 100 bigger! We investigated where this loss of accuracy came from. Figure 4 shows the relative error for the computed solution using the inversion formula (full line), while the dashed line indicates the relative error between $B_R(z)$ computed by the look-ahead procedure and $B_{R,E}(z)$ computed using (2) ($h_{\max} = 2$). In the left figure, the elements of the Toeplitz matrix were taken as random numbers uniformly distributed in $(0, 1)$ (as we always did up till now) while on the right-hand side, from $(-1, 1)$. Note that the difference between the relative errors for $B_R(z)$ and $V(z)$ is smaller for the $(-1, 1)$ case than for the $(0, 1)$ case. Hence, we concentrated on the $(0, 1)$ case. It turned out that applying the inversion formula led to the loss of two decimal digits. If we use the notation

$$\begin{aligned} V_1 &= L_{(n,\eta)}(X)U_{(\eta,m)}(W)Y \\ V_2 &= L_{(n,\eta)}(Z)U_{(\eta,m)}(U)Y, \end{aligned}$$

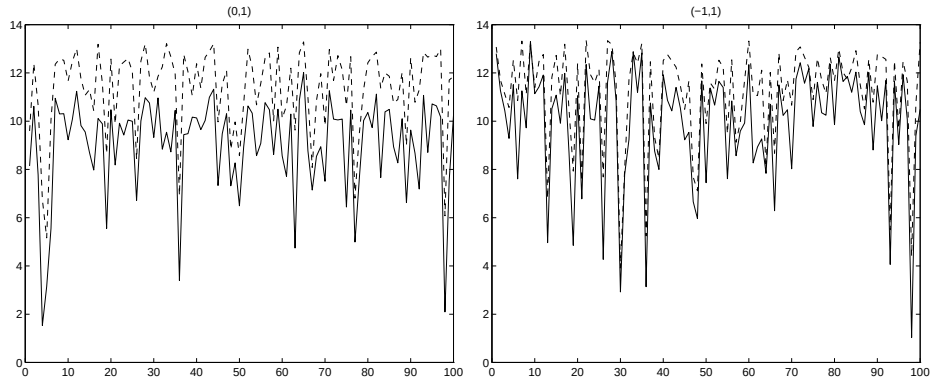


Figure 4: $-\log_{10}$(relative error) for $B_R(z)$ (dashed line) and $V(z)$ (full line)

we get for one of the 100 test cases (with similar results for the others) $||V_1||_2 = 497.6939$ and $||V_2||_2 = 498.3299$ while V is computed as the sum $V_1 + V_2 = V \approx [1, 1, \ldots, 1]^T$, so $||V||_2 \approx 8$. This explains the loss of about 2 decimal places.

In the next section, another method is given to compute the solution of a block Toeplitz system without using the inversion formula while in Section 15 the look-ahead criterion is refined. These two enhancements will lead to results comparable to those of Example 1 in [10]. We get even better results applying one step of iterative refinement using the inversion formula. Iterative improvement or iterative refinement is explained in [17, p. 126-127]. The stability of inversion formulas for scalar Toeplitz matrices is studied in [22].

14. Updating of the solution of a block Toeplitz system

The block Toeplitz system of linear equations

$$TV = Y$$

at point (m, n) can be written as a nonhomogeneous (right) interpolation problem.

DEFINITION 24. [nonhomogeneous interpolation problem] Let $Y(z) \in \mathbb{R}[z]^{p\times 1}$ and $T(z) \in \mathbb{R}[z]^{p\times q}$ be given. For each normal point $\nu = (m, n)$ of

$T(z)$, we call $V(z) \in \mathbb{R}[z]^{q\times 1}$ the solution of the nonhomogeneous interpolation problem iff

$$T(z)V(z) = Y(z) + O_-(z^{-1}) + O_+(z^m)$$

and $\deg V(z) < n$.

Note that for a normal point, the solution always exists and is unique.

We construct now the updating formulas to go from the solution $V(z)$ at the normal point $\nu = (m, n)$ to the solution $V'(z)$ at the normal point $\nu' = (m', n')$. Let us try to find the polynomial vectors $E(z) \in \mathbb{R}[z]^{p\times 1}$ and $F(z) \in \mathbb{R}[z]^{q\times 1}$ such that

$$V'(z) = V(z) - [X(z) \quad Z(z)] \begin{bmatrix} E(z) \\ F(z) \end{bmatrix}. \tag{21}$$

Because the degree of $V'(z)$ should be smaller than n', we limit the degrees of $E(z)$ and $F(z)$, i.e. $\deg E(z) < \delta_n$ and $\deg F(z) < \delta_n$ with $\delta_n = n' - n$. The nonhomogeneous interpolation conditions should be satisfied, i.e.

$$\begin{aligned} Y(z) &= T(z)V'(z) + O_-(z^{-1}) + O_+(z^{m'}) \\ &= T(z)V(z) - T(z)[X(z) \quad Z(z)] \begin{bmatrix} E(z) \\ F(z) \end{bmatrix} + O_-(z^{-1}) + O_+(z^{m'}) \end{aligned}$$

Because $T(z)V(z) = Y(z) + O_-(z^{-1}) + O_+(z^m)$ and $T(z)[X(z) \quad Z(z)] = [R_x^-(z) + R_x^+(z) \quad R_z^-(z) + R_z^+(z)]$, we find the coefficients e_i of $E(z)$ and f_i of $F(z)$ from the following system of linear equations with the coefficient matrix $S_{\nu,\nu'}$ as in (5)

$$S_{\nu,\nu'} \begin{bmatrix} e_0 \\ e_1 \\ \vdots \\ e_{\delta-1} \\ f_0 \\ f_1 \\ \vdots \\ f_{\delta_n-1} \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \\ R_{v,m}^+ - Y_m \\ R_{v,m+1}^+ - Y_{m+1} \\ \vdots \\ R_{v,m'-1}^+ - Y_{m'-1} \end{bmatrix} \tag{22}$$

with $T(z)V(z) = R_v^+(z) + O_-(z^{-1}) = R_{v,0}^+ + R_{v,1}^+ z^1 + \cdots + O_-(z^{-1})$ and $Y(z) = Y_0 + Y_1 z^1 + \cdots$. In Section 16, we give the results of the error analysis for the updating formula (21). In the next section, we refine the look-ahead criterion and show the accuracy of the solution computed using the updating formula compared to iterative refinement based on the inversion formula.

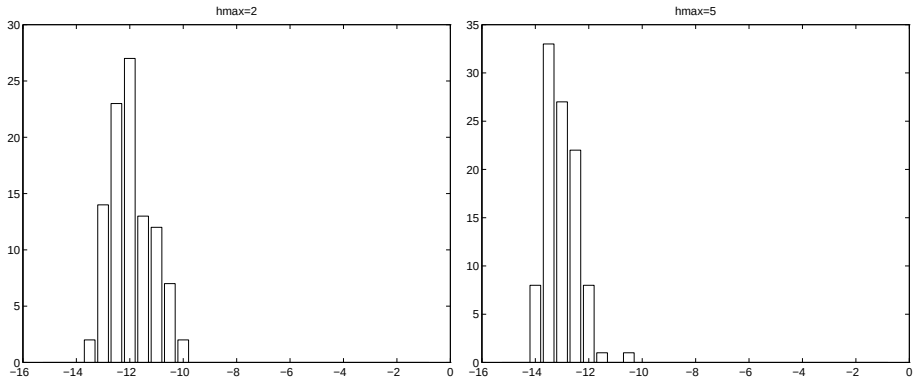


Figure 5: Histogram of $\log_{10}$(relative error)

15. Refinement of the look-ahead criterion

Instead of comparing κ with a fixed threshold τ, like we did in Algorithm 16, we compared κ with a value $\bar{\kappa}$ which was adapted during the execution of the algorithm. However, the value $\bar{\kappa}$ could only increase. This led in some cases to bad results. Therefore, if we find within the next $h_{\max}$ steps a κ which satisfies $\kappa < 2\bar{\kappa}$, we set the new value of $\bar{\kappa}$ equal to this κ if $\kappa < \bar{\kappa}$.

EXAMPLE 25. If we take hundred 64×64 scalar Toeplitz matrices with elements randomly distributed within $(-1, +1)$ and at least one of the principal submatrices is ill-conditioned, Figure 5 shows the effect of using the new look-ahead criterion and the updating formula (21) on the histogram of the relative errors for two different values of $h_{\max}$, 2 and 5. Figure 6 compares the old look-ahead criterion (dashed line) with the new one (full line) plotting the $-\log_{10}$ of the relative errors for the two values of $h_{\max}$, 2 and 5. Figure 7 plots the $-\log_{10}$ of the relative errors using the updating formula (21) (full line) compared to using the inversion formula (dashed line) and one step of iterative refinement (dash-dotted line) for $h_{\max} = 2$.

EXAMPLE 26. If we take hundred 210×210 block Toeplitz matrices $T_{M,N}$ with $p = 2$, $q = 3$, $M = 105$ and $N = 70$ with elements randomly distributed within $(-1, +1)$ and at least one of the principal block-submatrices is ill-conditioned, Figure 8 shows the effect on the histogram of the relative errors when using the new look-ahead criterion and the updating for-

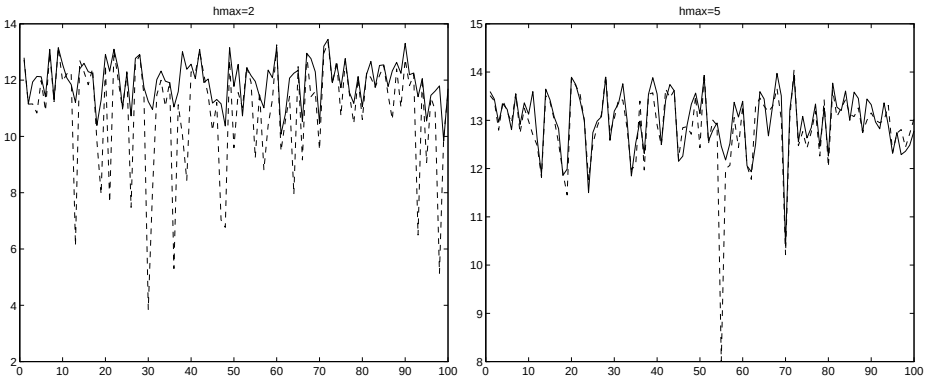


Figure 6: $-\log_{10}$(relative error) for the old (dashed) and the new look-ahead criterion(full line)

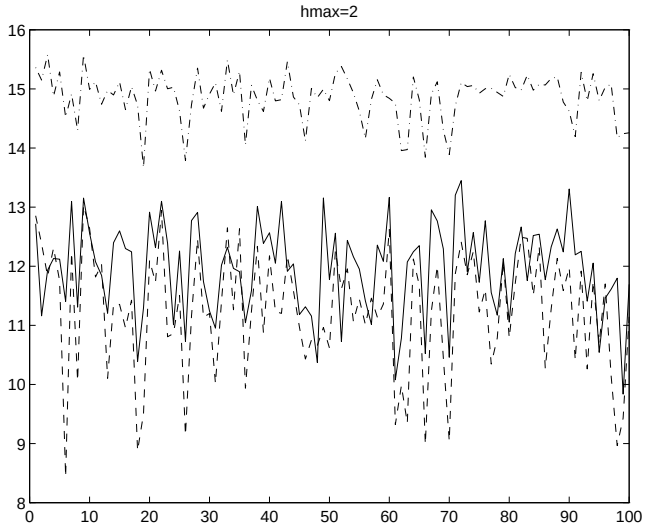


Figure 7: $-\log_{10}$(relative error) using the updating formula (full line) compared to the inversion formula (dashed line) and one step of iterative refinement (dash-dotted line)

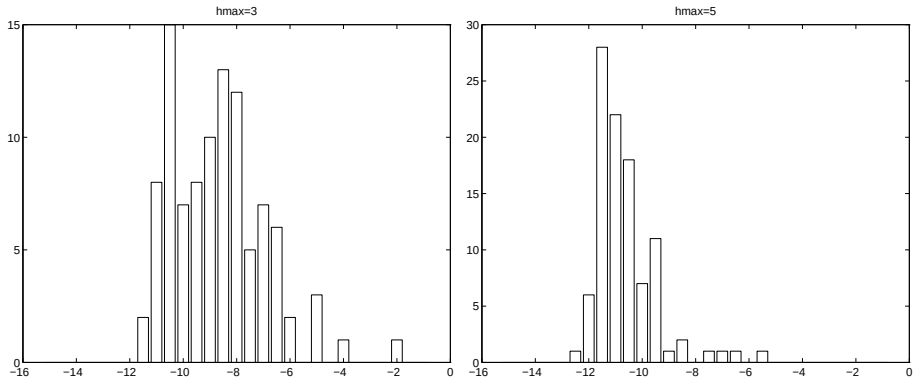


Figure 8: Histogram of $\log_{10}$(relative error)

mula (21). Figure 9 compares the old and the new look-ahead criterion. Figure 10 plots the $-\log_{10}$ of the relative errors using the updating formula (21) (dash-dotted line) compared to using the inversion formula (dashed line) and two steps of iterative refinement (full line) for $h_{\mathbf{max}} = 5$. Note that if a look-ahead step of size $h_{\mathbf{max}} = 5$ is taken, the $S_{\nu,\nu'}$ matrix is 30×30. For one of the matrices the $\log_{10}$(relative error) using the updating formula is between -6 and -4. Figure 11 shows $-\log_{10}$(relative error) of $B_R(z)$ and $V(z)$ and the $\log_{10}$ of the condition number of the corresponding principal submatrix in each of the normal points taken as look-ahead steps. This example shows that the look-ahead criterion does not always give satisfying results. Moreover, in the right part of Figure 12, we show that the relative error for the basis matrix (dash-dotted line) and the relative error for the computed solution (full line) of the block Toeplitz system of equations can have a different behaviour. These cases are rare. In the left part of Figure 12 we show the typical behaviour. For both examples $p = q = 1$, $M = N = 64$, $h_{\mathbf{max}} = 5$ and the elements are uniformly random in $(0, 1)$. Note that till now we have not taken into account the error analysis for the updating of the solution $V(z)$ in the look-ahead strategy. This will be done in the next section.

EXAMPLE 27. To measure the computational complexity, Figure 13 shows the number of multiplications of Gaussian elimination with back-substitution relative to the number of multiplications of the look-ahead method with one (or two steps) of iterative refinement. We get the same qualitative results when also taking into account the number of additions. In the left part of the figure, $p = q = 1$, $h_{\mathbf{max}} = 2$ and $M = N$ varies from

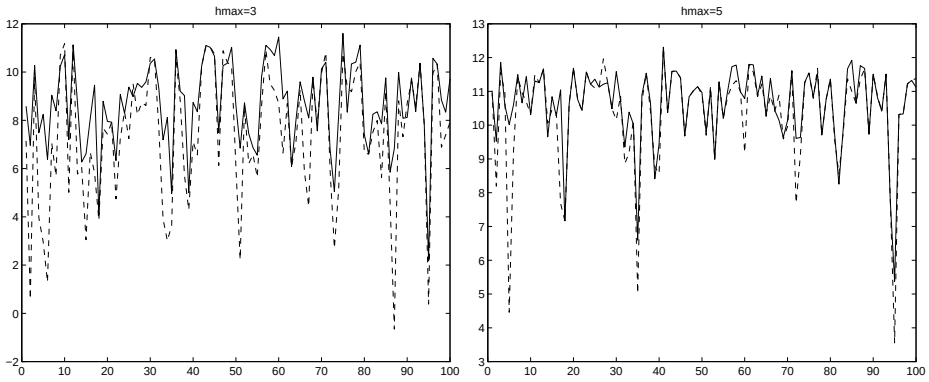


Figure 9: $-\log_{10}$(relative error) for the old (dashed) and the new look-ahead criterion(full line)

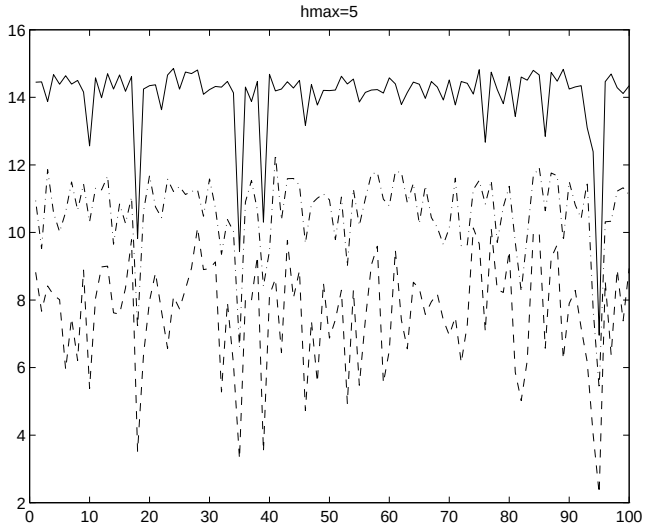


Figure 10: $-\log_{10}$(relative error) using the updating formula (dash-dotted line) compared to the inversion formula (dashed line) and one step of iterative refinement (full line)

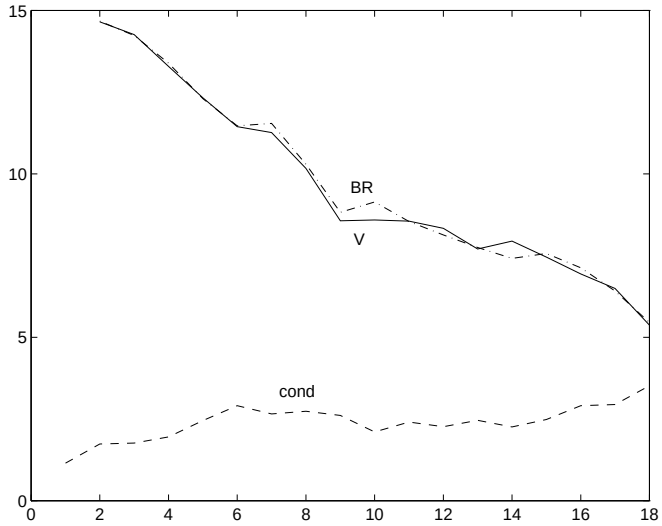


Figure 11: $-\log_{10}$(relative error) and $\log_{10}$(condition number)

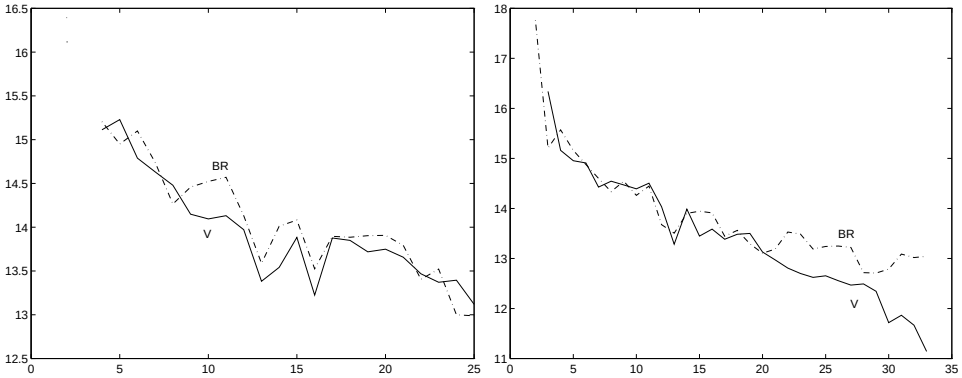


Figure 12: $-\log_{10}$(relative error) of $B_R(z)$ and of $V(z)$

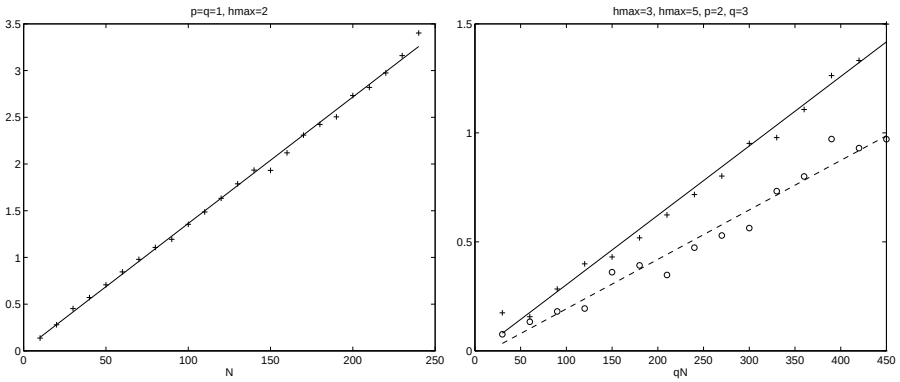


Figure 13: The number of multiplications of Gaussian elimination relative to the look-ahead method

10 to 240 with one step of iterative refinement. We have also taken into account that $X(z) = U(z)$ and $Z(z) = W(z)$. In the right part, $p = 2$, $q = 3$, $h_{\max} = 3$ (+) and $h_{\max} = 5$ (o) and N varies from 10 to 150 while two steps of iterative refinement are applied. Also the least square polynomial fit of degree 1 is shown. For the scalar case, this straight line is given by $0.0135N + 0.01$ while for the block case and $h_{\max} = 3$ the function values are $0.0032(qN) - 0.0149$ and for $h_{\max} = 5$ we get $0.0023(qN) - 0.0341$. From this fit, we obtain that Gaussian elimination needs more multiplications for the left part of the figure when $N > 73$ and for the right part when $qN > 319$ for $h_{\max} = 3$ and $qN > 456$ for $h_{\max} = 5$.

16. Error analysis

The computed solution $V(z)$ at point $\nu = (m^{(\sigma)}, n^{(\sigma)})$ does not satisfy the nonhomogeneous interpolation conditions exactly, i.e.

$$T(z)V^{(\sigma)}(z) = Y^{(\sigma)}(z) + \delta Y^{(\sigma)}(z) + O_-(z^{-1}) + O_+(z^{m^{(\sigma)}}).$$

We show how $\delta Y^{(\sigma)}(z)$ changes in each step σ. If it is clear from the context, we drop the index (σ).

THEOREM 28. *The error* $\delta Y^{(\sigma+1)}(z)$ *satisfies*

$$\begin{aligned}\delta Y^{(\sigma+1)}(z) &= \delta Y^{(\sigma)}(z) + \Pi_{0:m^{(\sigma+1)}-1}\big(-\theta_{XIII}(z) \\ &\quad -\theta_{XIV}(z) + T(z)\theta_{XV}(z) + (\delta R^{(\sigma)}(z) - \theta_V(z))EF(z)\big)\end{aligned}$$

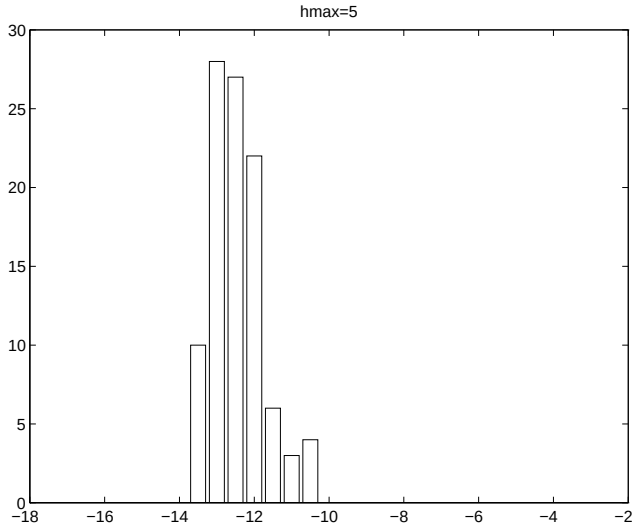


Figure 14: Histogram of $\log_{10}$(relative error)

with

$$\begin{aligned} ||\theta_V(z)|| &\leq 1.01(n+q)\mu \\ ||\theta_{XIII}(z)|| &\leq \mu(||R_j^+(z)|| + ||Y(z)||) \\ ||\theta_{XIV}(z)|| &\leq (8p^3(\delta_m+\delta_n)^3\rho_\sigma\mu + O(\mu^2))||EF(z)|| \\ ||\theta_{XV}(z)|| &\leq \mu||V(z)|| + 1.01\mu(\delta_n+p+q)||EF(z)|| \\ EF(z) &= [E(z)^T \quad F(z)^T]^T. \end{aligned}$$

Proof. See Theorem 17.5 of [28]. ■

Looking at the upper bound for the error $\delta Y^{(\sigma+1)}(z)$ we see that this error is additive over the different steps. The norm $||EF(z)||$ appears in several of the terms of the upper bound. We tried to design a look-ahead strategy in which the magnitude of $||EF^{(\sigma)}(z)||$ is compared to the magnitude of $||Y^{(\sigma+1)}(z)||$. For some single cases this gives an improvement. In general the results are a little bit worse than when using just the look-ahead criterion based on the κ's.

EXAMPLE 29. We take the 100 matrices of Example 25 and $h_{\max} = 5$ together with the new look-ahead criterion. Figure 14 should be compared to the right part of Figure 5.

17. Conclusion

In this paper, we have designed an algorithm for computing basis matrices for certain left and right interpolation problems where the data come from a block Toeplitz matrix. This algorithm is weakly stable and efficient. We have shown that the use of the inversion formula gives results which in almost all cases are not as good as using an updating formula for the solution of the block Toeplitz system of linear equations. However, one (or two) steps of iterative refinement using the inversion formula gives better accuracy than the updating formula. Hence, the best choice is to use the updating formula to compute an initial guess of the solution which is usually better than using the inversion formula and then execute one (or more) steps of iterative refinement using the inversion formula. This leads to an efficient and weakly stable method for solving a block-Toeplitz system of linear equations. However, we have given examples for which the method does not work that well. It is not clear for the moment where the look-ahead strategy fails in these rare cases.

We could have given another implementation by updating in each step besides the whole $(+)$-residual series also the whole $(-)$-residual series. In the implementation we developed in this paper, we only computed that part of the $(-)$-series needed to construct the $S_{\nu,\nu'}$ matrix.

REFERENCES

1 A.W. Bojanczyk and G. Heinig. A multi-step algorithm for Hankel matrices. *Journal of Complexity*, 10(1):142–164, March 1994.

2 S. Cabay, A. Jones, and G. Labahn. Computation of numerical Padé-Hermite and simultaneous Padé systems I: Near inversion of generalized Sylvester matrices. *SIAM J. Matrix Anal. Appl.*, 17:248–267, 1996.

3 S. Cabay, A. Jones, and G. Labahn. Computation of numerical Padé-Hermite and simultaneous Padé systems II: A weakly stable algorithm. *SIAM J. Matrix Anal. Appl.*, 17:268–297, 1996.

4 S. Cabay, A.R. Jones, and G. Labahn. A stable algorithm for multidimensional Padé systems and the inversion of generalized Sylvester matrices. Technical Report TR 94-07, Department of Computing Science, The University of Alberta, Edmonton, Alberta, Canada, March 1994.

5 S. Cabay, A.R. Jones, and G. Labahn. Experiments with a weakly stable algorithm for computing Padé-Hermite and simultaneous Padé

approximants. *ACM Trans. Math. Software*, pages 1–30, 1996. Submitted.

6 S. Cabay and R. Meleshko. A weakly stable algorithm for Padé approximants and the inversion of Hankel matrices. *SIAM J. Matrix Anal. Appl.*, 14:735–765, 1993.

7 T. F. Chan and Per C. Hansen. A look-ahead Levinson algorithm for general Toeplitz systems. *IEEE Trans. Sig. Proc.*, 40(5):1079–1090, 1992.

8 T. F. Chan and Per C. Hansen. A look-ahead Levinson algorithm for indefinite Toeplitz systems. *SIAM J. Matrix Anal. Appl.*, 13(2):490–506, 1992.

9 R.W. Freund. A look-ahead Bareiss algorithm for general Toeplitz matrices. *Numer. Math.*, 68:35–69, August 1994. AT&T Bell Labs. Numerical Analysis Manuscript 93-11.

10 R.W. Freund and H. Zha. Formally biorthogonal polynomials and a look-ahead Levinson algorithm for general Toeplitz systems. *Linear Algebra Appl.*, 188/189:255–303, 1993.

11 R.W. Freund and H. Zha. A look-ahead strategy for the solution of general Hankel systems. *Numer. Math.*, 64:295–322, 1993.

12 K. Gallivan, S. Thirumalai, and P. Van Dooren. A look-ahead Schur algorithm. In *Fifth SIAM Conference on Applied Linear Algebra, Snowbird, Utah*, pages 450–454, June 1994.

13 K. Gallivan, S. Thirumalai, P. Van Dooren, and V. Vermaut. High performance algorithms for Toeplitz and block Toeplitz matrices. *Linear Algebra Appl.*, 241-243:343–388, 1996.

14 K. Gallivan, S. Thirumalai, and P. Van Dooren. A block Toeplitz look-ahead Schur algorithm. In M. Moonen and B. De Moor, editors, *SVD and Signal Processing III*, pages 199–206, Amsterdam, 1995. Elsevier.

15 I. Gohberg, T. Kailath, and V. Olshevsky. Fast Gaussian elimination with partial pivoting for matrices with displacement structure. *Math. Comp.*, 64(212):1557–1576, 1995.

16 I. Gohberg and T. Shalom. On Bezoutians of nonsquare matrix polynomials and inversion of matrices with nonsquare blocks. *Linear Algebra Appl.*, 137/138:249–323, 1990.

17 G.H. Golub and Ch.F. Van Loan. *Matrix Computations.* The Johns Hopkins University Press, Baltimore and London, second edition edition, 1989.

18 M. Gutknecht and M. Hochbruck. Optimized look-ahead recurrences for adjacent rows in the Padé table. *BIT*, 36:264–286, 1996.

19 M.H. Gutknecht. Stable row recurrences for the Padé table and generically superfast lookahead solvers for non-Hermitian Toeplitz systems. *Linear Algebra Appl.*, 188/189:351–422, 1993.

20 M.H. Gutknecht and M. Hochbruck. Look-ahead Levinson- and Schur-type recurrences in the Padé table. *Electron. Trans. Numer. Anal.*, 2:104–129, 1994.

21 M.H. Gutknecht and M. Hochbruck. Look-ahead Levinson and Schur algorithms for non-Hermitian Toeplitz systems. *Numer. Math.*, 70:181–228, 1995.

22 M.H. Gutknecht and M. Hochbruck. The stability of inversion formulas for Toeplitz matrices. *Linear Algebra Appl.*, 223/224:307–324, 1995.

23 G. Heinig. Transformation approaches for fast and stable solution of Toeplitz systems and polynomial equations. In *Proceedings of the International Workshop "Recent advances in Applied Mathematics"*, pages 223–238, Kuwait, May 1996.

24 G. Heinig and A. Bojanczyk. Transformation techniques for Toeplitz and Toeplitz-plus-Hankel matrices I. Transformations. *Linear Algebra Appl.*, pages 1–24, 1996. To appear.

25 G. Heinig and A. Bojanczyk. Transformation techniques for Toeplitz and Toeplitz-plus-Hankel matrices II. Algorithms. *Linear Algebra Appl.*, pages 1–20, 1996. To appear.

26 T. Huckle. A look-ahead algorithm for solving nonsymmetric linear Toeplitz equation. In *Fifth SIAM Conference on Applied Linear Algebra, Snowbird, Utah*, pages 455–459, June 1994.

27 T. Kailath and A. Sayed. Displacement structure: Theory and applications. *SIAM Rev.*, 37(3):297–386, 1995.

28 M. Van Barel and A. Bultheel. A look-ahead algorithm for the solution of block Toeplitz systems. Technical Report TW224, Department of Computer Science, K.U. Leuven, March (Revised May, September) 1995. (34 pages).