SIAM J. MATRIX ANAL. APPL.
Vol. 12, No. 3, pp. 497–520, July 1991

008

IMMITANCE-TYPE THREE-TERM SCHUR AND LEVINSON RECURSIONS FOR QUASI-TOEPLITZ COMPLEX HERMITIAN MATRICES*

Y. BISTRITZ†‡, H. LEV-ARI†, AND T. KAILATH†

Abstract. A comprehensive analysis is made of Schur- and Levinson-type algorithms for Toeplitz and quasi-Toeplitz matrices that have half the number of multiplications and the same number of additions as the classical algorithms. Several results of this type have appeared in the literature under the label "split algorithms." In this approach the reduction in computation is obtained by a two-step procedure: (i) the first step is a variable (or "recursion-type") transformation from the classical (i.e., "scattering") variables to a new (so-called, "immitance") set of variables, which by itself reduces the number of multiplications at the cost of increasing the number of additions; (ii) the second step achieves control of the number of additions by converting the two-term recursions into the lesser known (for discrete orthogonal polynomials) three-term recursions. In the Toeplitz case the new variables turn out to be the odd and even parts of the classical variables, leading to the terminology of split algorithms, but this feature is lost in the quasi-Toeplitz case. Nevertheless, the network-theoretic interpretation of a transformation from scattering to immittance variables can still be maintained. Certain judicious choices of free parameters have to be made in each case in order to achieve the maximum computational reduction. It is shown how these results yield efficient procedures for determining the inertia of a quasi-Toeplitz matrix and the location of roots of its "predictor" polynomials from the immittance-type three-term recursions. In particular, connections with the Bistritz stability test, which was the motivation for our study of the Levinson and Schur algorithms in this paper, are noted.

Key words. Levinson algorithm, Schur algorithm, Toeplitz matrices, fast immittance-type recursions

AMS(MOS) subject classifications. primary 65F05, 65F30; secondary 15A06

1. Introduction. Several recent papers have introduced computationally efficient (three-term) versions of the well-known Levinson algorithm and the somewhat less well known Schur algorithm. Bistritz has obtained several tests [1]–[4] for the root distribution of polynomials with respect to the unit circle that involve only the even (or odd) parts of the polynomials, and needed only half the number of multiplications (and the same number of additions) as the well-known Schur–Cohn test [5]. Since the Schur–Cohn test is essentially a reverse (degree-reducing) form of the Levinson algorithm, as well as a particular case of the Schur algorithm, it was reasonable to expect that similar reductions in computational complexity could also be obtained for both the Levinson and the Schur algorithms. Indeed, Delsarte and Genin derived one such computationally improved version for both of these algorithms: in [6] and [7] they presented the so-called "split Levinson" and "split Schur" algorithms for symmetric Toeplitz matrices with *real entries*. The adjective "split" arises from the ability to work with the odd and even (or symmetric and skew-symmetric) parts of the polynomials involved in the usual Levinson algorithm. Such improved algorithms were also obtained, in a slightly different context, by Bube and Burridge [23]. In our previous work [8], [9] we proved that: (i) the same approach applies not only to Toeplitz but also to certain *quasi-Toeplitz* (or Bezoutian) matrices, where the polynomials in the improved Levinson algorithm are not symmetric or skew-symmetric and *cannot* be viewed as an even/odd split of the polynomials in the usual

* Received by the editors September 4, 1987; accepted for publication (in revised form) March 6, 1990. This research was supported in part by the Air Force Office of Scientific Research, Air Force Systems Command contract AF-83-0228 and by U.S. Army Research Office contract DAALO3-86-K-0045.

† Information Systems Laboratory, Stanford University, Stanford, California 94305 (bistritz@genius.tau.ac.il, levari@northeastern.edu, and kailath@isl.stanford.edu).

‡ Present address, Department of Electrical Engineering, Tel-Aviv University, Tel-Aviv 69978, Israel. This author's research was supported in part by a Chaim Weizmann postdoctoral fellowship award.

Levinson algorithm, and (ii) there are *three* computationally efficient and different-in-form versions of the Levinson algorithm. Finally, we note that the extension of the Bistritz stability test to the complex case (i.e., to polynomials with complex coefficients) was derived independently by Delsarte, Genin, and Kamp [10] and by Bistritz [2] and that Krishna and Morgera [11], [12] were perhaps the first to publish complex versions of the split Levinson algorithm.

This paper explores in detail the possibility of reducing the computational complexity of both the Levinson and the Schur algorithms by studying the effects of *variable* or *recursion-type transformations* and of introducing *three-term recursions*. We believe that the comparison of alternative computational procedures should not be carried out solely in terms of their computational requirements: other attributes, such as numerical robustness or suitability for parallel implementation, may be more significant in certain applications. Therefore, we consider in this paper *all* $O(N^2)$ *alternatives* to the conventional (scattering-type, two-term) recursions. Having established an explicit characterization of all efficient alternatives of the Schur and Levinson algorithms, we are in a position to *prove* that one so-called balanced immittance-type three-term version of the recursions (coinciding with the recursion in [11] and [12] in the Toeplitz case) has the lowest computational requirements. This advantage of the balanced version over all other alternatives has not been established in previous publications (i.e., in [6], [7], [11], [12], [20]), because no comparison to other alternatives was available. Moreover, we also prove that *all efficient three-term equivalents of the Schur/Levinson recursions are related to each other by scaling*.

We present our results in the somewhat generalized context of *quasi-Toeplitz matrices with complex entries*. We do so not only for the sake of extending results otherwise known for Toeplitz matrices, but mainly to establish the fact that the structural form of the recursions and the reduction in computational requirements depend not upon the special (persymmetry) property of Toeplitz matrices, but instead upon their so-called displacement structure. In contrast, the approaches used in previous publications rely heavily on the persymmetry property.

Before proceeding to a more specific outline of the background and the contributions of this paper we may suggest that the reader might also find it useful to scan the remarks in the concluding section of the paper.

1.1. The Levinson algorithm for quasi-Toeplitz matrices. The Levinson algorithm is a fast method that recursively solves, for $n = 1, \cdots, N$, the set of linear equations

$$[a_{n,n} \cdots a_{n,1} \quad 1]\mathbf{R}_{0:n} = R_n^e[0 \cdots 0 \quad 1] \tag{1a}$$

for the unknowns $\{a_{n,i}, R_n^e\}$, where $\mathbf{R}_{0:n}$ is the $(n+1) \times (n+1)$ leading submatrix of $\mathbf{R}_{0:N}$. The system matrix $\mathbf{R}_{0:N}$ is either a square Hermitian *Toeplitz* matrix, say

$$\mathbf{T}_{0:N} = \{c_{i-j}; 0 \leqq i, j \leqq N\}, \tag{1b}$$

or a square Hermitian *quasi-Toeplitz* matrix, i.e., one of the form

$$\mathbf{R}_{0:N} = \mathbf{H}\mathbf{T}_{0:N}\mathbf{H}^*, \tag{1c}$$

where $\mathbf{H}$ is a lower-triangular Toeplitz matrix of size $(N+1) \times (N+1)$, and the asterisk $(^*)$ denotes Hermitian transpose (complex conjugate for scalars). An alternative characterization of quasi-Toeplitz matrices is that they have *displacement inertia* (1, 1), as defined in [13], viz., $\mathbf{R}_{0:N}$ is such that the displacement matrix $\mathbf{R}_{0:N} - \mathbf{Z}\mathbf{R}_{0:N}\mathbf{Z}^*$ has one positive eigenvalue and one negative eigenvalue (and $N+1-2$ zero eigenvalues), where

$\mathbf{Z}$ is a matrix with unity elements on the first subdiagonal and zeros elsewhere. This means that

$$\mathbf{R}_{0:N} - \mathbf{Z}\mathbf{R}_{0:N}\mathbf{Z}^* = \mathbf{u}_0\mathbf{u}_0^* - \mathbf{v}_0\mathbf{v}_0^* \tag{1d}$$

for some column vectors $\mathbf{u}_0$, $\mathbf{v}_0$. For notational simplicity in further analysis we shall scale the matrix $\mathbf{R}_{0:N} = \{r_{i,j};\ 0 \leqq i, j \leqq N\}$ so that its top-left element is

$$r_{0,0} = 1. \tag{1e}$$

Therefore, in particular, Toeplitz matrices must satisfy $c_0 = 1$ and, consequently, the lower-triangular Toeplitz matrix $\mathbf{H}$ in (1c) must have unity diagonal elements. We may note that for certain choices of $\{\mathbf{u}_0, \mathbf{v}_0\}$ the matrix $\mathbf{R}_{0:N}$ becomes a so-called unit-circle Bezoutian, familiar from stability theory (see the discussion at the end of § 5).

Following [14], we say that $\mathbf{R}_{0:N}$ is *admissible* if there exists a scalar ρ such that

$$\mathbf{u}_0 - \rho\mathbf{v}_0 = [1 \quad 0 \cdots 0]^T. \tag{1f}$$

If $\mathbf{R}_{0:N}$ is admissible, then it is always possible to choose $\mathbf{v}_0(z)$ so that the corresponding ρ is real and nonnegative. We should also emphasize that by varying $\mathbf{H}$ in (1c), we obtain a *family* of quasi-Toeplitz matrices $\mathbf{R}_{0:N}$, all sharing the same reflection coefficients $\{k_n\}$. Some of these quasi-Toeplitz matrices are admissible and can be completely characterized by specifying the scalar coefficient $\rho \geqq 0$; others are nonadmissible and require a specification of $N + 1$ additional coefficients (see [14]).

Equation (1a) can be solved via the (generalized Levinson) recursions [14]

$$\begin{pmatrix} a_n(z) \\ b_n(z) \end{pmatrix} = L_n(z)\begin{pmatrix} a_{n-1}(z) \\ b_{n-1}(z) \end{pmatrix}, \qquad L_n(z) = \begin{pmatrix} z & -k_n \\ -k_n^* z & 1 \end{pmatrix}, \tag{2a}$$

where

$$a_n(z) := \sum_{i=0}^{n} a_{n,i} z^{n-i}, \tag{2b}$$

$b_n(z)$ is an auxiliary polynomial with coefficients $b_{n,i}$, viz.,

$$b_n(z) := \sum_{i=0}^{n} b_{n,i} z^i, \tag{2c}$$

and

$$a_0(z) = 1, \qquad b_0(z) = \rho. \tag{2d}$$

If $\mathbf{R}_{0:N}$ is not admissible, then its Levinson recursion is a further generalization of (2), which we shall not discuss in this paper, but which is indicated in [14]. For Toeplitz matrices, $\rho = 1$ and the recursions (2) become the well-known Levinson–Szegö recursions for the *orthogonal polynomials* $a_n(z)$ [15], with $b_n(z) \equiv a_n^{\#}(z) := z^n[a_n(z^{-*})]^*$, the *conjugate reverse polynomial of* $a_n(z)$. The *reflection coefficients* k_n are computed by certain inner-product formulas, which we discuss in further detail in § 4. We also recall here the readily verified fact that, by stacking the solutions of (1a) for $i = 0, 1, \cdots, N$, we can get the unique upper-diagonal-lower (UDL) triangular factorization of the inverse

of $\mathbf{R}_{0:N}$, $\mathbf{R}_{0:N}^{-1} = \mathbf{A}_{0:N}^{*}\mathbf{D}_{0:N}^{-1}\mathbf{A}_{0:N}$, where $\mathbf{D}_{0:N} = \text{diag}\{R_i^e;\ 0 \leqq i \leqq N\}$ and $\mathbf{A}_{0:N}$ is a lower-triangular matrix whose nth row contains the coefficients of $a_n(z)$, viz.,

$$\mathbf{A}_{0:N} = \begin{bmatrix} 1 & & & \\ a_{1,1} & 1 & & \\ \vdots & \vdots & \ddots & \\ a_{N,N} & a_{N,N-1} & \cdots & 1 \end{bmatrix}. \tag{3}$$

1.2. The Schur algorithm for quasi-Toeplitz matrices. The Schur algorithm is an alternative (and more direct) method for computing the reflection coefficients, which at the same time also determines the unique lower-diagonal-upper (LDU) triangular factorization of the matrix $\mathbf{R}_{0:N}$ itself, rather than its inverse [16]. It involves a recursion that we can rearrange (see Appendix B in [9]) in a form that is identical to the Levinson recursion (2a), viz.,

$$\begin{pmatrix} \tilde{u}_n(z) \\ \tilde{v}_n(z) \end{pmatrix} = L_n(z) \begin{pmatrix} \tilde{u}_{n-1}(z) \\ \tilde{v}_{n-1}(z) \end{pmatrix} \tag{4a}$$

where $u_n(z)$, $v_n(z)$ are power-series in z, viz.,

$$u_n(z) = \sum_{i=0}^{N} u_{n,i} z^i, \qquad v_n(z) = \sum_{i=0}^{N} v_{n,i} z^i, \tag{4b}$$

and $\tilde{u}_n(z)$ denotes conjugation of coefficients alone in the power series $u_n(z)$, i.e.,

$$\tilde{u}_n(z) := [u_n(z^*)]^*. \tag{4c}$$

Admissibility is not involved at all in the Schur recursion (4), which can be applied to every quasi-Toeplitz matrix.

The recursion starts with $u_0(z)$, $v_0(z)$. The coefficients $u_{0,i}$, $v_{0,i}$ of these polynomials are the elements of the column vectors $\mathbf{u}_0$, $\mathbf{v}_0$ in the displacement representation (1d) for $\mathbf{R}_{0:N}$. The representation (1d) of $\mathbf{R}_{0:N}$ is nonunique, as we can replace, for instance, the two-column matrix $[\mathbf{u}_0\ \mathbf{v}_0]$ by $[\mathbf{u}_0\ \mathbf{v}_0]\Theta(k)$, where

$$\Theta(k) := \frac{1}{\sqrt{1-|k|^2}} \begin{pmatrix} 1 & -k \\ -k^* & 1 \end{pmatrix}.$$

In particular, we can always select $\mathbf{u}_0$, $\mathbf{v}_0$ such that the first element of $\mathbf{v}_0$ vanishes, i.e.,

$$\mathbf{u}_0 := \begin{pmatrix} 1 \\ u_{0,1} \\ \vdots \\ u_{0,N} \end{pmatrix}, \qquad \mathbf{v}_0 := \begin{pmatrix} 0 \\ v_{0,1} \\ \vdots \\ v_{0,N} \end{pmatrix}, \tag{5}$$

where we use the convention (1e) that $r_{0,0} = 1$. In particular, for a Toeplitz matrix,

$$u_0(z) = \sum_{i=0}^{N} c_i z^i, \qquad v_0(z) = u_0(z) - 1, \tag{6}$$

which satisfies the constraint (5) with $u_{0i} = v_{0i}$ for $i > 0$. Moreover, the recursion (4a) imposes the same constraint upon all subsequent $v_n(z)$, i.e., $v_{n,n} = 0$ for all n. Note that, in addition, the first n coefficients of both $u_n(z)$ and $v_n(z)$ always equal zero.

The LDU factorization of $\mathbf{R}_{0:N}$ is obtained as follows: the nth diagonal element of the diagonal matrix $\mathbf{D}_{0:N}$ in $\mathbf{R}_{0:N} = \mathbf{L}_{0:N}\mathbf{D}_{0:N}\mathbf{L}_{0:N}^{*}$ is

$$d_n = u_{n,n} = R_n^e = \prod_{i=1}^{n} (1 - |k_i|^2), \qquad R_0^e = 1, \tag{7}$$

and $\mathbf{L}_{0:N}$ is a lower triangular matrix that has the coefficients of $u_n(z)/d_n$ as the elements in the nth column. The last equality in (7) is well known for Toeplitz matrices (see, e.g., [15]). In fact, it holds also for quasi-Toeplitz matrices because the $\{R_n^e\}$ corresponding to the quasi-Toeplitz matrix $\mathbf{R}_{0:N}$ of (1a), (1c) are independent of the matrix $\mathbf{H}$ and coincide with the $\{R_n^e\}$ that would appear in equations (1c) with the Toeplitz matrix $\mathbf{T}_{0:N}$. This is so because the lower-triangular matrix $\mathbf{H}$ must have unity diagonal elements in order to conform with the scaling convention (1e).

The computational costs of the Schur and Levinson algorithms are similar (see Table 2 for a summary of operation counts). The Schur algorithm is, however, more advantageous for *parallel* computation because it does not involve inner products (see, e.g., [16]). We shall now see how further computational reductions can be achieved for both algorithms.

1.3. Variable transformations and three-term recursions. Our approach to the problem of reducing computational requirements is different from that of Delsarte and Genin [6], [7] and that of Krishna and Morgera [11], [12], and it follows the method used in [8] and [9][1]: first we make a suitable variable transformation and then we convert the resulting two-term recursion into a three-term form. Thus consider first a linear transformation of the recursions, viz.,

$$\begin{pmatrix} f_n(z) \\ g_n(z) \end{pmatrix} := T_n \begin{pmatrix} a_n(z) \\ b_n(z) \end{pmatrix}, \qquad \begin{pmatrix} x_n(z) \\ y_n(z) \end{pmatrix} := T_n \begin{pmatrix} \tilde{u}_n(z) \\ \tilde{v}_n(z) \end{pmatrix}, \tag{8a}$$

which results in a modified set of two-term recursions. Namely,

$$\begin{pmatrix} f_n(z) \\ g_n(z) \end{pmatrix} = T_n L_n(z) T_{n-1}^{-1} \begin{pmatrix} f_{n-1}(z) \\ g_{n-1}(z) \end{pmatrix} \tag{8b}$$

and similarly for the Schur recursion. Note that the effect of the (nonsingular) matrices T_n is to transform the degree one polynomial matrices $L_n(z)$ into another set of matrices of the same nature. Thus, the modified recursions (8b) require $O(N^2)$ operations *for every choice of the transformation matrices* $\{T_n;\ 0 \leqq n \leqq N\}$.

An alternative form of the recursions is obtained by eliminating $g_n(z)$ altogether from (8b). This results in a three-term recursion, i.e., $f_n(z)$ is determined from $f_{n-1}(z)$ and $f_{n-2}(z)$, rather than from $f_{n-1}(z)$ and $g_{n-1}(z)$. The three-term version of the recursion may, in general, involve *polynomial division*, which significantly raises the computational requirements. We show in § 2 that the only way to avoid this additional computation is to choose

$$T_n = \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix} \begin{pmatrix} \psi_n & 0 \\ 0 & \nu_n \end{pmatrix}, \tag{9a}$$

[1] It might be noted that the work in [6], [7], [11], and [12] deals only with the Toeplitz case, for which the reduction is obtained by working with the symmetric (or skew-symmetric) parts of the polynomial $a_n(z)$, leading to the name "split Levinson," introduced in [6]. However, in the quasi-Toeplitz case this symmetry is not available, though equivalent computational reductions can still be obtained.

where $\{\psi_n, \nu_n\}$ are complex scalars, and the ratio

$$\eta_n := \nu_n/\psi_n \tag{9b}$$

is constrained by the recursion

$$\eta_n = \frac{\eta_{n-1} + k_n}{1 + k_n^* \eta_{n-1}}, \qquad \eta_0 = 1. \tag{9c}$$

Since this recursion completely determines η_n in terms of the reflection coefficients $\{k_i;\ 1 \leqq i \leqq n\}$ and since $f_n(z) = \psi_n[a_n(z) + \eta_n b_n(z)]$, it follows that all efficient three-term equivalents of the Schur/Levinson recursions are related to each other by scaling (note that the η_i all have unit modulus). A suitable choice of the scaling coefficients $\{\psi_n\}$ may reduce the number of nontrivial coefficients in the recursion. As will be seen in § 2, the most efficient version involves a single complex multiplication[2] per recursion step, per coefficient. This unique, computationally efficient, version has the form

$$f_{n+1}^B(z) = (\delta_n z + \delta_n^*) f_n^B(z) - z f_{n-1}^B(z). \tag{10}$$

For real covariances, $\delta_n = \delta_n^*$, and this recursion reduces to the so called *balanced* recursion of [8] and [9]. Previous publications pointed out that scaling the $f_n(z)$ polynomials produces $O(N^2)$ equivalents of the balanced recursion, but did not show that every three-term immittance-type equivalent of the balanced recursion is produced in this manner.

It turns out that, in addition to the balanced recursion, there are only four distinct versions of the recursion with *two nontrivial coefficients*. They consist of two pairs: the monic/comonic pair and the dual-codual pair. The *monic/comonic recursions* have the form

$$f_{n+1}^M(z) = \left(z + \frac{\eta_n}{\eta_{n-1}}\right) f_n^M(z) - \lambda_n z f_{n-1}^M(z), \tag{11a}$$

$$f_{n+1}^{CM}(z) = \left(\frac{\eta_{n-1}}{\eta_n} z + 1\right) f_n^{CM}(z) - \lambda_n^* z f_{n-1}^{CM}(z), \tag{11b}$$

and reduce, for real covariances, to the monic recursion of [8] and [9]. The *dual/codual recursions* have the form

$$\lambda_{n+1} f_{n+1}^D(z) = \left(z + \frac{\eta_n}{\eta_{n-1}}\right) f_n^D(z) - z f_{n-1}^D(z), \tag{12a}$$

$$\lambda_{n+1}^* f_{n+1}^{CD}(z) = \left(\frac{\eta_{n-1}}{\eta_n} z + 1\right) f_n^{CD}(z) - z f_{n-1}^{CD}(z) \tag{12b}$$

and reduce, for real covariances, to the dual recursion of [8] and [9]. Though more computationally expensive than (10), we introduce (11)–(12) for completeness and because they may have other applications (e.g., (10)–(12) may have different degrees of numerical robustness).

---

[2] More precisely, the equivalent of a single complex multiplication, i.e., a total of four real multiplications. We remark also that in the real case we have three distinct versions of the recursion with a single nontrivial coefficient.

Of course, the recursions (10)–(12) also hold for $x_n(z)$ in the Schur algorithm. For instance, the balanced recursion for $x_n(z)$ is

$$x^B_{n+1}(z) = (\delta_n z + \delta_n^*) x^B_n(z) - z x^B_{n-1}(z),$$

and similarly for the other four versions.

The analysis in this paper extends the results of [8] and [9], including the useful *transmission line* interpretation. The ratio $v_n(z)/u_n(z)$ in the Schur algorithm is bounded by unity (for $|z| < 1$) and can be interpreted as the scattering function of a transmission line consisting of a cascade of (uniform) sections with different characteristic impedances. On the other hand, the ratio $x_n(z)/y_n(z)$ is positive-real (for $|z| < 1$) and can be interpreted as the impedance (or admittance) function of a related transmission line. For this reason we shall say that the recursions (2), (4) are of the *scattering type*, whereas the transformed recursions (i.e., those for (f_n, g_n) or for (x_n, y_n)) are of the *immittance type*.[3] Indeed, if we denote $s_n(z) := \tilde{v}_n(z)/\tilde{u}_n(z)$, then $c_n(z) := y_n(z)/x_n(z)$ is given by

$$c_n(z) = [1 - \eta_n s_n(z)]/[1 + \eta_n s_n(z)],$$

which we recognize as the well-known Cayley transform, mapping bounded functions into positive-real functions and vice versa.

The derivation of the three-term immittance-type recursions (10)–(12) is carried out in § 2. In order to propagate these recursions, beginning with the given covariance $\mathbf{R}_{0:N}$, we must also have formulas for computing the coefficients $\{\lambda_i, \delta_i\}$, similar to those used in the scattering-type formulation of the Levinson and the Schur algorithms to compute the reflection coefficients $\{k_i\}$; these calculations, which require the same number of multiplications as in the scattering-type recursions, are derived in §§ 3 and 4. We also present, in § 4, the relations required to reconstruct $\{g_n(z)\}$ from the three-term recursion for $\{f_n(z)\}$; this makes it possible to reconstruct the predictor polynomials $\{a_n(z)\}$ when necessary. Finally, § 5 briefly considers the relation between the immittance-type parameters $\{\delta_n\}$ and the *inertia* (i.e., the number of positive, null, and negative eigenvalues) of quasi-Toeplitz matrices. In particular, we show that any quasi-Toeplitz matrix $\mathbf{R}_{0:N}$ is congruent to a *tridiagonal* (Jacobi) matrix ∇_N whose nontrivial elements are the parameters $\{\delta_n\}$ (see (49)). Consequently, both matrices have the same inertia. This congruence relationship also appears in recent work of Delsarte and Genin (see, e.g., [20]). We present, in § 5, an efficient computational procedure for determining the inertia of ∇_N. We also show how to apply this procedure to locate the roots of the polynomial $a_N(z)$ of (2a) with respect to the unit circle.

2. Transformed recursions and three-term forms. We introduced in [8] and [9] the general linear transformation (8a), viz.,

$$\begin{pmatrix} f_n(z) \\ g_n(z) \end{pmatrix} := T_n \begin{pmatrix} a_n(z) \\ b_n(z) \end{pmatrix}, \tag{13}$$

where T_n is any constant nonsingular 2×2 matrix. This results in a transformed two-term recursion for $f_n(z)$, $g_n(z)$, namely,

$$\begin{pmatrix} f_n(z) \\ g_n(z) \end{pmatrix} = \begin{pmatrix} \alpha_n(z) & \beta_n(z) \\ \gamma_n(z) & \delta_n(z) \end{pmatrix} \begin{pmatrix} f_{n-1}(z) \\ g_{n-1}(z) \end{pmatrix}, \tag{14a}$$

[3] Bode coined the term *immittance* to denote both *impedance* and *admittance* [17].

where

$$\begin{pmatrix} \alpha_n(z) & \beta_n(z) \\ \gamma_n(z) & \delta_n(z) \end{pmatrix} := T_n \begin{pmatrix} z & -k_n \\ -k_n^* z & 1 \end{pmatrix} T_{n-1}^{-1}, \tag{14b}$$

from which we can obtain a three-term recursion for $f_n(z)$. The same transformation can be applied to the Schur recursions (4). The corresponding *transformed Schur recursions are obtained by replacing* here (and in the remainder of § 2) $a_n(z)$, $b_n(z)$ by $\tilde{u}_n(z)$, $\tilde{v}_n(z)$ and, similarly, $f_n(z)$, $g_n(z)$ by $x_n(z)$, $y_n(z)$.

Since the three-term recursion for $f_n(z)$ does not involve $g_n(z)$, it should not depend upon the elements in the second row of the transformation matrix T_n. We may, therefore, assume any particular form for the second row of T_n, for instance,

$$T_n := \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix} \begin{pmatrix} \psi_n & 0 \\ 0 & \nu_n \end{pmatrix}, \qquad \psi_n \neq 0 \neq \nu_n \tag{15a}$$

without affecting at all the three-term recursion for $f_n(z)$. Thus,

$$f_n(z) = \psi_n [a_n(z) + \eta_n b_n(z)], \tag{15b}$$

where

$$\eta_n := \nu_n / \psi_n. \tag{15c}$$

When the underlying covariances are real-valued, the choice $\eta_n = 1$ leads to the simplest recursions, which we have already analyzed in [9].

Following the general technique for converting two-term recursions into three-term recursions (see [9]) we obtain

$$f_{n+1}(z) = \left\{ \alpha_{n+1}(z) + \frac{\beta_{n+1}(z)\delta_n(z)}{\beta_n(z)} \right\} f_n(z) - \frac{\beta_{n+1}(z)\Delta_n(z)}{\beta_n(z)} f_{n-1}(z), \tag{16a}$$

where

$$\Delta_n(z) := \alpha_n(z)\delta_n(z) - \gamma_n(z)\beta_n(z) = \frac{\psi_n \nu_n}{\psi_{n-1}\nu_{n-1}}(1 - |k_n|^2)z. \tag{16b}$$

Also,

$$g_n(z) = \frac{\delta_n(z) f_n(z) - \Delta_n(z) f_{n-1}(z)}{\beta_n(z)}. \tag{16c}$$

Since $\alpha_n(z)$, $\beta_n(z)$, $\gamma_n(z)$, $\delta_n(z)$ are all polynomials of degree one (see explicit expressions in Appendix A), it follows that the three-term recursion involves *rational coefficients*, which significantly complicates the computation. Thus a simplified three-term recursion for $f_n(z)$ is possible if and only if these coefficients become polynomials in z of degree one, or less. We show in Appendix A that the rational coefficient $\beta_{n+1}(z)/\beta_n(z)$ becomes a constant if and only if

$$\frac{\eta_{n+1} - k_{n+1}}{1 - \eta_{n+1} k_{n+1}^*} = \eta_n, \tag{17a}$$

which can also be written as an ascending recursion in η_n, viz.,

$$\eta_{n+1} = \frac{\eta_n + k_{n+1}}{1 + \eta_n k_{n+1}^*}, \qquad \eta_0 := 1. \tag{17b}$$

Our choice of the initial condition $\eta_0 = 1$ is motivated by the observation that for real reflection coefficients $\{k_n\}$, this choice leads to $\eta_n = 1$ for all n, while for complex reflection coefficients it still yields

$$|\eta_n| = 1. \tag{17c}$$

With the ratio $\eta_n = \nu_n/\psi_n$ being constrained as in (17), the three-term recursion simplifies to (see Appendix A for derivation)

$$f_{n+1}(z) = \frac{\psi_{n+1}}{\psi_n}\frac{\eta_{n+1} - k_{n+1}}{\eta_n - k_n}\left\{\left(\frac{\eta_{n-1}}{\eta_n}z + 1\right)f_n(z) - \frac{\psi_n}{\psi_{n-1}}(1 - |k_n|^2)zf_{n-1}(z)\right\} \tag{18a}$$

and the auxiliary expression for $g_n(z)$ becomes

$$(z-1)g_n(z) = (\zeta_n z + \zeta_n^*)f_n(z) - \frac{\psi_n}{\psi_{n-1}}(1 - \eta_n k_n^*)(\zeta_n + \zeta_n^*)zf_{n-1}(z), \tag{18b}$$

where

$$\zeta_n := \frac{1 + \eta_n k_n^*}{1 - \eta_n k_n^*}. \tag{18c}$$

These expressions reduce, for $\eta_n = 1$ (for real k_n), to (16a), (16b) in [8] and [9].

To initialize the three-term recursion for $\{f_n(z)\}$, one needs to know both $f_0(z)$ *and* $f_1(z)$. The definition (8), combined with the two-term recursion (2), implies that

$$f_0(z) = \psi_0\{a_0(z) + b_0(z)\}, \qquad f_1(z) = \psi_1(\eta_1 - k_1)\{za_0(z) + b_0(z)\}, \tag{19}$$

but it will be more convenient to have the initial conditions on $f_{-1}(z)$ and $f_0(z)$. Both (18a) and (18b) imply that

$$zf_{-1}(z) = \frac{\psi_{-1}}{1 - |k_0|^2}\{(1 + \eta_{-1}k_0^* z)a_0(z) + \eta_{-1}(\eta_{-1}z + k_0)b_0(z)\}.$$

These initial conditions involve the undefined quantities ψ_0, ψ_{-1}, η_{-1}, and k_0. We show in Appendix B that a consistent choice for these quantities is

$$\psi_0 = (1 - k_0)^{-1}, \qquad \psi_{-1} = 1 + k_0, \qquad \eta_{-1} = 1, \tag{20}$$

where k_0 is not subject to any constraints. In particular, if we choose $k_0 = -1$, then

$$\psi_0 = \tfrac{1}{2}, \qquad \psi_{-1} = 0, \qquad \eta_{-1} = 1, \tag{21a}$$

which results in

$$2zf_{-1}(z) = (1 - z)\{a_0(z) - b_0(z)\}, \qquad 2f_0(z) = a_0(z) + b_0(z). \tag{21b}$$

This choice is motivated by the observation that the Levinson recursions for Toeplitz matrices are initialized with $a_0(z) = 1 = b_0(z)$, which reduces (21b) to $z\,f_{-1}(z) = 0$ and $f_0(z) = 1$.

A further reduction in complexity can be achieved by appropriately choosing the scaling factors ψ_n. There is a single choice that leads to recursions with *one* nontrivial coefficient, and four choices that lead to recursions with *two* nontrivial coefficients:

(1) *Balanced recursion*, obtained by choosing ψ_n to satisfy the constraints

$$\frac{\psi_{n+1}^B}{\psi_{n-1}^B}\frac{\eta_{n+1} - k_{n+1}}{\eta_n - k_n}(1 - |k_n|^2) = 1, \qquad n \geq 0 \tag{22a}$$

and

$$(\psi_n^B)^* = \eta_n \psi_n^B, \qquad n \geq 0, \tag{22b}$$

resulting in

$$f_{n+1}^B(z) = (\delta_n z + \delta_n^*) f_n^B(z) - z f_{n-1}^B(z), \qquad n \geq 0. \tag{23}$$

The special form of the multiplier of $f_n^B(z)$ with

$$\delta_n = \frac{\psi_{n+1}^B(\eta_{n+1} - k_{n+1})\eta_{n-1}}{\psi_n^B(\eta_n - k_n)\eta_n}$$

is established in Appendix B. As a consequence of the initialization (20), $\delta_0 = (1 - |k_0|^2)^{-1}(\psi_{-1}/\psi_0)^* = 1$. The remaining $\{\delta_n\}$ are related to each other by a recursion derived from the constraint on ψ_n^B, viz.,

$$\delta_n \delta_{n-1} \lambda_n = 1, \qquad n \geq 1, \qquad \delta_0 = 1, \tag{24a}$$

where (note that $\eta_n^* = \eta_n^{-1}$ by (17c))

$$\lambda_n := (\eta_{n-1} + k_n)(\eta_{n-1} - k_{n-1})^*, \qquad n \geq 1. \tag{24b}$$

The reason for the name "balanced" for (23) is that the recursions for ascending and descending indices are, essentially, identical.

Note that the balanced recursions involve only four real multiplications (i.e., the equivalent of one complex multiplication) per recursion step, per coefficient. In fact, the balanced recursion can be carried out as two interlacing three-term recursions that involve only real arithmetic [2]. Decomposing $f_n(z)$ into two real polynomials, viz.,

$$f_n(z) = S_n(z) + jA_n(z) \tag{25}$$

and separating the real and imaginary parts of (23), we obtain

$$S_{k+1}(z) = \delta_k^I(z+1)S_k(z) + \delta_k^R(z-1)A_k(z) - zS_{k-1}(z), \tag{26a}$$

$$A_{k+1}(z) = \delta_i^I(z+1)A_k(z) - \delta_k^R(z-1)S_k(z) - zA_{k-1}(z), \tag{26b}$$

where δ_k^R and δ_k^I denote the real and imaginary parts of δ_k, respectively. The recursions (26) involve four real multiplications and eight real additions per recursion step, per polynomial coefficient. In the Toeplitz case, $S_n(z)$ and $A_n(z)$ are, respectively, symmetric and skew-symmetric, and only half of their coefficients need to be computed.

(2) *Monic recursion*, obtained by choosing ψ_n to satisfy the constraint

$$\psi_n^M(1 - \eta_n k_n^*) = 1, \qquad n \geq 0 \tag{27}$$

resulting in

$$f_{n+1}^M(z) = \left(z + \frac{\eta_n}{\eta_{n-1}}\right) f_n^M(z) - \lambda_n z f_{n-1}^M(z), \qquad n \geq 0. \tag{28}$$

Note that as a consequence of the initialization (20), $\lambda_0 = \psi_0(1 - |k_0|^2)/(\eta_{-1}\psi_{-1}) = 1$ also. This recursion involves *two complex multiplications* per recursion step, per coefficient, and is therefore, in general, inferior to the balanced recursion. The reason for the name "monic" for (28) is that, with the appropriate initialization, $\{f_n^M(z)\}$ in the Levinson recursion are monic polynomials.

(3) *Comonic recursion*, obtained by choosing ψ_n to satisfy the constraint

$$\psi_n^{CM}(\eta_n - k_n) = 1, \qquad n \geq 0 \tag{29}$$

resulting in the comonic recursion (11b), which has the same computational complexity as the monic recursion (28). The reason for the name "comonic" is that the $f_n^{CM}(z)$ in the Levinson recursion for *Toeplitz matrices* are comonic polynomials. However, the same property does not hold for other (quasi-Toeplitz) matrices.

(4) *Dual recursion*, obtained by choosing ψ_n to satisfy the constraint

$$\eta_n \psi_n^D (1 - |k_n|^2) = \eta_{n-1} \psi_{n-1}^D, \qquad n \geqq 0 \tag{30}$$

resulting in

$$\lambda_{n+1} f_{n+1}^D(z) = \left(z + \frac{\eta_n}{\eta_{n-1}} \right) f_n^D(z) - z f_{n-1}^D(z), \qquad n \geqq 0. \tag{31}$$

This recursion also involves *two complex multiplications* per recursion step, per coefficient.

(5) *Codual recursion*, obtained by choosing ψ_n to satisfy the constraint

$$\psi_n^{CD}(1 - |k_n|^2) = \psi_{n-1}^{CD}, \qquad n \geqq 0 \tag{32}$$

resulting in the codual recursion (12b), which has the same computational complexity as the dual recursion (31).

Remark. Note that, in view of (15b), $f_n^B(z)$, $f_n^M(z)$, $f_n^{CM}(z)$, $f_n^D(z)$, $f_n^{CD}(z)$ are all proportional to $a_n(z) + \eta_n b_n(z)$ and, therefore, to each other. The coefficients of proportionality can be determined by comparing the leading coefficients in these polynomials. Since $f_n^M(z)$ is monic, it follows, for instance, that $f_n^B(z)/f_n^M(z) = \prod_{i=0}^{n-1} \delta_i$. We show in Appendix B that

$$f_n^M(z) = \xi_{n-1} f_n^B(z), \qquad f_n^{CM}(z) = \xi_{n-1}^* f_n^B(z), \tag{33a}$$

$$f_n^D(z) = \xi_n^{-1} f_n^B(z), \qquad f_n^{CD}(z) = \xi_n^{-*} f_n^B(z), \tag{33b}$$

where

$$\xi_n := \prod_{i=0}^{n} \delta_i^{-1}. \tag{33c}$$

The same proportionality coefficients also relate the various versions of $x_n(z)$ in the immittance-type Schur algorithm.

The recursions just described are incomplete because we have not given methods for computing the coefficients λ_n, δ_n in them from the given matrix $\mathbf{R}_{0:N}$. There are two generic methods of doing this—what we call the Schur-type, where these coefficients are computed as certain ratios, and the Levinson-type, where their computation involves certain inner products. Besides the fact that Schur-type algorithms are better adapted to parallel computation, we also note that the functions propagated in the Schur-type recursions yield the Cholesky factors of $\mathbf{R}_{0:N}$, while those in the Levinson-type recursion yield the factors of $\mathbf{R}_{0:N}^{-1}$. We could also use the Schur recursions to compute the coefficients and then, under the assumption of admissibility, use the coefficients to compute the polynomials in the Levinson recursions.

3. Immittance-type Schur algorithms. We first review the Schur method for computing the scattering-type reflection coefficients $\{k_n\}$. From (4) we note that since $[z^{-n} v_n(z)]_{z=0} = 0$, it follows that k_n is the ratio of two known coefficients, viz.,

$$k_n = \left. \frac{v_{n-1}(z)}{z u_{n-1}(z)} \right|_{z=0} = \frac{v_{n-1,n}}{u_{n-1,n-1}}. \tag{34}$$

TABLE 1
Immittance-type three-term Schur recursions for real covariances.

Balanced	Monic	Dual
$zx_{-1}(z) = \frac{1}{2}(1-z)[u_0(z) - v_0(z)]$		
$x_0(z) = \frac{1}{2}[u_0(z) + v_0(z)]$		
$\delta_0 = 1 = \lambda_0$		
for $n = 0, 1, 2, \cdots, N-1$ **do**		
$\delta_n = [zx^B_{n-1}(z)/x^B_n(z)]\|_{z=0}$ (*)	$\lambda_n = [x^M_n(z)/zx^M_{n-1}(z)]\|_{z=0}$ (*)	$\tilde{x}^D_{n+1}(z) := (z+1)x^D_n(z) - zx^D_{n-1}(z)$
$x^B_{n+1}(z) = \delta_n(z+1)x^B_n(z) - zx^B_{n-1}(z)$	$x^M_{n+1}(z) = (z+1)x^M_n(z) - \lambda_n zx^M_{n-1}(z)$	$\lambda_{n+1} = [z^{-(n+1)}\tilde{x}^D_{n+1}(z)]\|_{z=0}$ $x^D_{n+1}(z) = \lambda^{-1}_{n+1}\tilde{x}^D_{n+1}(z)$

(*) Skip this step for $n = 0$.

A similar approach can be used for the three-term immittance-type Schur recursion and it yields, for instance, the expression $\delta_n^* = [zx^B_{n-1}(z)/x^B_n(z)]_{z=0}$. Combining such expressions with the recursions and initial conditions (21)–(32), we obtain a family of complete Schur algorithms, which we summarize below.

3.1. Real covariances. The analysis of § 2 yields *three* computationally efficient sets of recursions, which are summarized in Table 1. Note that all three versions begin with the same initial conditions $x_{-1}(z)$, $x_0(z)$. Also, all three versions require a *single real multiplication and two real additions* per recursion per each coefficient of $x_{n+1}(z)$, as compared to *two real multiplications and two real additions* for the scattering-type Schur algorithm (4) for real quasi-Toeplitz covariances.

3.2. Complex covariances. The analysis of § 2 yields a *single* computationally efficient recursion (the balanced version), viz.,

$$x^B_{n+1}(z) = (\delta_n z + \delta_n^*)x^B_n(z) - zx^B_{n-1}(z), \qquad n \geq 0, \tag{35a}$$

where

$$x^B_0(z) := \tfrac{1}{2}\{\tilde{u}_0(z) + \tilde{v}_0(z)\}, \qquad zx^B_{-1}(z) := \tfrac{1}{2}(1-z)\{\tilde{u}_0(z) - \tilde{v}_0(z)\} \tag{35b}$$

and, with the notation $x^B_n(z) = \sum_{i=0}^N x^B_{n,i}z^i$,

$$\delta_n^* := \left.\frac{zx^B_{n-1}(z)}{x^B_n(z)}\right|_{z=0} = \frac{x^B_{n-1,n-1}}{x^B_{n,n}}. \tag{35c}$$

We emphasize again that even though (35a) seems to involve two nontrivial coefficients, namely, δ_n and δ_n^*, it can be carried out by two real three-term recursions, similar to (26), and requires, in fact, only *four real multiplications and eight real additions* per recursion step per each coefficient of $x^B_{n+1}(z)$. This is half the number of multiplications and the same number of additions as compared to the scattering-type Schur algorithm (4) for complex covariances. The relative efficiency of the immittance-type algorithm over the scattering-type one is, therefore, the same for both real and complex covariances and for all quasi-Toeplitz matrices (see Table 2).

TABLE 2
Computation counts.

	Scattering		Immittance	
	Mult.	Add.	Mult.	Add.
Schur				
Real	$O(N^2)$	$O(N^2)$	$O(0.5N^2)$	$O(N^2)$
Complex	$O(4N^2)$	$O(4N^2)$	$O(2N^2)$	$O(4N^2)$
Levinson: quasi-Toeplitz				
Real	$O(1.5N^2)$	$O(1.5N^2)$	$O(N^2)$	$O(1.5N^2)$
Complex	$O(6N^2)$	$O(6N^2)$	$O(4N^2)$	$O(6N^2)$
Levinson: Toeplitz				
Real	$O(N^2)$	$O(N^2)$	$O(0.5N^2)$	$O(N^2)$
Complex	$O(4N^2)$	$O(4N^2)$	$O(2N^2)$	$O(4N^2)$

We use the notation $m = O(aN^p)$ to mean that m is a polynomial of degree p in N, viz., $m = aN^p + bN^{p-1} + \cdots$.

4. Immittance-type Levinson algorithms. The scattering-type Levinson recursions (2) involve the reflection coefficients $\{k_n\}$, which are usually computed via an inner-product formula[4]

$$k_n = \frac{1}{R^e_{n-1}} \mathbf{a}_{n-1}[c_1 \cdots c_n]^T, \tag{36}$$

where $\mathbf{a}_{n-1} := [a_{n-1,n-1} \cdots a_{n-1,1} \quad 1]$ is a row vector consisting of the coefficients of the polynomial $a_{n-1}(z)$, and R^e_n is updated by

$$R^e_n = (1 - |k_n|^2) R^e_{n-1}, \qquad R^e_0 = c_0 = 1. \tag{37}$$

Similarly, the immittance-type Levinson recursions (10)–(12) involve the recursion coefficients $\{\delta_n\}$, $\{\lambda_n\}$, which can also be computed via suitable inner-product formulas, as we presently show.

Let $\mathbf{R}_{0:N}$ be the quasi-Toeplitz covariance associated with the pair $\mathbf{u}_0$, $\mathbf{v}_0$ via (5). If this covariance is *admissible*, i.e., if $u_0(z) = 1 + \rho v_0(z)$ for some scalar ρ, then, as we have shown in Appendix B of [9] (see also [22]),

$$\mathbf{a}_n[1 \quad u_{0,1} \cdots u_{0,n}]^T = 0, \tag{38a}$$

$$\mathbf{b}_n[1 \quad u_{0,1} \cdots u_{0,n}]^T = \rho R^e_n \tag{38b}$$

for all $n \geq 1$, where $\mathbf{a}_n$, $\mathbf{b}_n$ are row vectors consisting of the coefficients of $a_n(z)$, $b_n(z)$, respectively. Consequently,

$$\tau_n = \psi_n \eta_n \rho R^e_n, \qquad n \geq 1, \tag{39a}$$

where

$$\tau_n := \mathbf{f}_n[1 \quad u_{0,1} \cdots u_{0,n}]^T \tag{39b}$$

[4] This expression for k_n has to be slightly modified for quasi-Toeplitz matrices (see [9]).

and $\mathbf{f}_n$ is the row vector consisting of the coefficients of the polynomial $f_n(z)$. We may extend (39a) to $n = 0$ and *define* τ_0 as

$$\tau_0 := \rho\psi_0\eta_0 R_0^e = \frac{\rho}{2}. \tag{39c}$$

It turns out that the recursion coefficients $\{\delta_n\}$, $\{\lambda_n\}$ can always be computed as ratios of subsequent τ_n. For instance, the coefficients of the balanced recursion (23) are given by (see Appendix B)

$$\delta_n := \frac{\psi_{n+1}}{\psi_n}\frac{\eta_{n+1}-k_{n+1}}{\eta_n-k_n}\frac{\eta_{n-1}}{\eta_n} = \frac{\psi_{n-1}\eta_{n-1}}{\psi_n\eta_n(1-|k_n|^2)},$$

which, by comparison with (39a), implies that

$$\delta_n = \frac{\tau_{n-1}^B}{\tau_n^B}, \qquad n \geqq 1. \tag{40}$$

Combining such expressions with the recursions and initial conditions (21)–(32) we obtain a family of complete Levinson algorithms, which we summarize below.

4.1. Real covariances. The analysis of § 2 yields *three* computationally efficient sets of recursions, which are summarized in Table 3. Note that all three versions begin with the same initial conditions $zf_{-1}(z) = \frac{1}{2}(1 - z)(1 - \rho)$, $f_0(z) = \frac{1}{2}(1 + \rho)$, which are the initial conditions presented in [9] (but differ from those in [8] in also allowing $\rho = 0$). All three recursions require a *single real multiplication and two real additions* per recursion per each coefficient of $f_{n+1}(z)$, as compared to *two real multiplications and two real additions* for the scattering-type Levinson algorithm for real quasi-Toeplitz covariances. The computation of the recursion coefficients via (39b) requires one inner-product and one division per recursion, which is the same as in the scattering-type Levinson algorithm. In the Toeplitz case the (conjugate) symmetry of the polynomial $f_n(z)$ results in a further reduction of the computational requirements (see Table 2).

TABLE 3
Immittance-type three-term Levinson recursions for real covariances.

Balanced	Monic	Dual
$zf_{-1}(z) = \frac{1}{2}(1-z)(1-\rho)$		
$f_0(z) = \frac{1}{2}(1+\rho)$		
$\tau_0 = \frac{1}{2}\rho, \qquad \delta_0 = 1 = \lambda_0$		
for $n = 0, 1, 2, \cdots, N-1$ **do**		
$f_{n+1}^B(z)$ $= \delta_n(z+1)f_n^B(z) - zf_{n-1}^B(z)$	$f_{n+1}^M(z)$ $= (z+1)f_n^M(z) - \lambda_n zf_{n-1}^M(z)$	$\tilde{f}_{n+1}^D(z)$ $:= (z+1)f_n^D(z) - zf_{n-1}^D(z)$
$\sum_i f_{n,i}^B z^i := f_n^B(z)$	$\sum_i f_{n,i}^M z^i := f_n^M(z)$	$\sum_i \tilde{f}_{n,i}^D z^i := \tilde{f}_n^D(z)$
$\tau_n^B = \sum_i f_{n,i}^B u_{0,i}$ (*)	$\tau_n^M = \sum_i f_{n,i}^M u_{0,i}$ (*)	$\lambda_{n+1} = 2\rho^{-1}\sum_i \tilde{f}_{n+1,i}^D u_{0,i}$
$\delta_n = \tau_{n-1}^B/\tau_n^B$ (*)	$\lambda_n = \tau_n^M/\tau_{n-1}^M$ (*)	$f_{n+1}^D(z) = \lambda_{n+1}^{-1}\tilde{f}_{n+1}^D(z)$

(*) Skip this step for $n = 0$.

4.2. Complex covariances. The analysis of § 2 yields a *single* computationally efficient recursion (the balanced version), viz.,

$$f_{n+1}^{B}(z)=(\delta_n z+\delta_n^{*})f_n^{B}(z)-zf_{n-1}^{B}(z), \qquad n\geqq 0, \tag{41a}$$

where

$$f_0^{B}(z):=\tfrac{1}{2}(1+\rho), \qquad zf_{-1}^{B}(z):=\tfrac{1}{2}(1-z)(1-\rho), \tag{41b}$$

and

$$\delta_n:=\frac{\tau_{n-1}^{B}}{\tau_n^{B}}, \qquad n\geqq 1, \tag{41c}$$

where τ_n is computed via the inner-product (39b), viz.,

$$\tau_n:=\mathbf{f}_n[1 \quad u_{0,1}\cdots u_{0,n}]^{T}, \qquad n\geqq 1, \qquad \tau_0=\frac{\rho}{2}. \tag{41d}$$

We emphasize that (41a) can be carried out by (26) and requires, like the corresponding Schur algorithm, only *four real multiplications and eight real additions* per recursion step per each coefficient of $f_{n+1}^{B}(z)$. This is half the number of multiplications and the same number of additions as compared to the scattering-type Levinson recursion for quasi-Toeplitz complex covariances. If the covariance matrix is not Toeplitz, the inner product formula has the same efficiency as in the scattering-type formulation for both the complex and real cases. Consequently, the relative efficiency of the immittance-type Levinson algorithms is the same for both real and complex quasi-Toeplitz covariances (a factor of 1.5, see Table 2) and is less than the (factor of two) relative efficiency of the corresponding Schur algorithms.

In the Toeplitz case, however, the symmetry in $f_n(z)$ can be exploited to simplify the computation of τ_n, viz.,

$$\begin{aligned}\tau_n=&[u_{0,0}+u_{0,n} \quad u_{0,1}+u_{0,n-1}\cdots u_{0,[n/2]}+u_{0,[(n+1)/2]}][S_{n,0} \quad S_{n,1}\cdots S_{n,[n/2]}]^{T}\\ &+j[u_{0,0}-u_{0,n} \quad u_{0,1}-u_{0,n-1}\cdots u_{0,[n/2]}-u_{0,[(n+1)/2]}][A_{n,0} \quad A_{n,1}\cdots A_{n,[n/2]}]^{T},\end{aligned} \tag{42}$$

where $S_n(z)$ and $A_n(z)$ are the real polynomials obtained from the real and imaginary parts of the coefficients of $f_n(z)$ (see (26)), and $[x]$ denotes the integer part of a real number x. If the Toeplitz covariance matrix is real, all $A_n(z)$ vanish, and the simplified formula (42) can be used for the three efficient versions of the Levinson algorithm for real Toeplitz matrices, as already mentioned in [8] and [9]. In summary, since both the inner-product formula (42) and the recursions (41a) have half the complexity of the corresponding scattering-type equivalents, the relative efficiency of the immittance-type algorithm is the same (i.e., a factor of 2) for both real and complex Toeplitz covariances, and is comparable to the corresponding efficiencies of the Schur algorithms (see Table 2).

4.3. Recovery of the orthogonal polynomials. The orthogonal polynomial $a_n(z)$ can always be recovered from $f_n(z)$ and $g_n(z)$ by inverting the recursion-type transformation (8), viz.,

$$a_n(z)=(2\psi_n)^{-1}\{f_n(z)+g_n(z)\},$$

which suggests the more convenient expression

$$a_n(z)=\frac{f_n(z)+g_n(z)}{f_n(\infty)+g_n(\infty)}, \tag{43}$$

where $f_n(\infty)$ indicates the leading coefficient of the polynomial $f_n(z)$.

In order to recover $g_n(z)$ from $\{f_i(z);\ 0 \leqq i \leqq n\}$, we observe that the balanced version of (18b) is

$$(z-1)g_n^B(z) = (\zeta_n z + \zeta_n^*) f_n^B(z) - 2\mu_n z f_{n-1}^B(z), \tag{44a}$$

where

$$\mu_n := \frac{1+\zeta_n}{2\delta_n}. \tag{44b}$$

This is so because

$$\delta_n^{-1} = \frac{\eta_n \psi_n^B}{\eta_{n-1}\psi_{n-1}^B}(1-|k_n|^2) = \frac{\psi_n^B}{\psi_{n-1}^B}\frac{1-\eta_n k_n^*}{1-\eta_n^* k_n}(1-|k_n|^2),$$

and combining this with the identity (A.5b) simplifies the coefficient of $z f_{n-1}^B(z)$ in (18b) to

$$\frac{\psi_n^B}{\psi_{n-1}^B}(1-\eta_n k_n^*)(\zeta_n+\zeta_n^*) = \frac{2}{\delta_n(1-\eta_n k_n^*)} = \frac{1+\zeta_n}{\delta_n}.$$

An alternative expression for $g_n(z)$ can be obtained by using the balanced recursion (41a) to eliminate $z f_{n-1}^B(z)$ from (44a). This results in

$$(z-1)g_n^B(z) = 2\mu_n f_{n+1}^B(z) - (z+1) f_n^B(z), \tag{45a}$$

which also implies that the coefficient μ_n can be computed directly via the expression

$$\mu_n = \frac{f_n^B(1)}{f_{n+1}^B(1)}. \tag{45b}$$

Note that the coefficient μ_n is *real*, even though both ζ_n and δ_n are, in general, complex. This follows from comparing (A.5a) with (B.3). As a consequence, the evaluation of $(z-1)g_n^B(z)$ via (45) involves a single multiplication of a complex-valued vector by a real scalar. In comparison, using (44) for the same purpose involves an additional multiplication of a complex-valued vector by a complex scalar. Furthermore, using (44) requires us to also compute ζ_n itself, in addition to μ_n. It follows from (45) and the balanced recursion (23) that

$$\frac{2\delta_n}{1+\zeta_n} = \mu_n^{-1} = 2\delta_n^R - \mu_{n-1}, \tag{46}$$

which provides the real multiplier in the right-hand side of (44) and also, since δ_n is known, gives ζ_n itself for (44).

5. Inertia and stability. As is well known, a Hermitian Toeplitz matrix is positive definite if and only if the magnitude of its reflection coefficients is strictly less than one. More generally, the inertia of a Hermitian Toeplitz matrix coincides with the inertia of the diagonal matrix $\mathbf{D}_{0:N} = \operatorname{diag}\{R_n^e;\ 0 \leqq n \leqq N\}$ and can therefore be conveniently determined from the reflection coefficients via the relation (7). The same holds for quasi-Toeplitz matrices because all matrices congruent to a common Toeplitz matrix (as in (10)) share the same inertia [14].

The *real* coefficients $\{\mu_n\}$ that were introduced in (44)–(46) contain the same information as the $\{R_n^e\}$, because (see Appendix B for proof)

$$\frac{\mu_0}{\mu_n} = 2\prod_{i=1}^{n} |\delta_i|^2 \cdot (1-|k_i|^2). \tag{47}$$

In other words, the matrices $\mathbf{R}_{0:N}$, diag $\{R_n^e;\ 0 \leqq n \leqq N\}$, and μ_0^{-1} diag $\{\mu_n;\ 0 \leqq n \leqq N\}$ are all congruent to each other and, consequently, have the same inertia. Therefore, in particular, a quasi-Toeplitz matrix is positive definite if and only if the ratios $\{\mu_n/\mu_0\}$ are all strictly positive. Notice that this result is independent of the choice of initial conditions.

The most convenient way to compute the $\{\mu_n\}$ coefficients is via the recursion (46), viz.,

$$\mu_n^{-1} = 2\delta_n^R - \mu_{n-1}, \tag{48}$$

which is the immittance-type "analogue" of the recursive relation $R_n^e = R_{n-1}^e(1 - |k_n|^2)$. This relation also implies that there is yet another matrix with the same inertia as $\mathbf{R}_{0:N}$. Indeed, (48) implies that

$$\begin{bmatrix} 1 & & & & \\ \mu_0 & 1 & & & \\ & \mu_1 & 1 & & \\ & & \ddots & \ddots & \\ & & & \mu_{N-1} & 1 \end{bmatrix} \begin{bmatrix} \mu_0^{-1} & & & \\ & \mu_1^{-1} & & \\ & & \ddots & \\ & & & \mu_N^{-1} \end{bmatrix} \begin{bmatrix} 1 & \mu_0 & & & \\ & 1 & \mu_1 & & \\ & & 1 & \ddots & \\ & & & \ddots & \mu_{N-1} \\ & & & & 1 \end{bmatrix}$$

$$= \begin{bmatrix} \mu_0^{-1} & 1 & & & \\ 1 & 2\operatorname{Re}\delta_1 & 1 & & \\ & 1 & \ddots & \ddots & \\ & & \ddots & \ddots & 1 \\ & & & 1 & 2\operatorname{Re}\delta_N \end{bmatrix} := \nabla_N, \tag{49}$$

which proves that (with $\mu_0 > 0$) the tridiagonal matrix ∇_N is congruent to diag $\{\mu_n^{-1};\ 0 \leqq n \leqq N\}$ and hence to $\mathbf{R}_{0:N}$.

We note that the principal minors of ∇_N are given by $\{\sigma_n;\ 0 \leqq n \leqq N\}$ where

$$\sigma_{-1} := 1, \qquad \sigma_0 = 2, \qquad \sigma_n = 2\delta_n^R \sigma_{n-1} - \sigma_{n-2}, \qquad n = 1, \cdots, N, \tag{50}$$

and by comparison with (48),

$$\mu_n = \sigma_{n-1}/\sigma_n. \tag{51}$$

Therefore, the inertia of $\mathbf{R}_{0:N}$ can be determined from the signs of the μ_n computed by (48) or from the sign *changes* in the σ_n sequence computed by (50). In conclusion, the matrix $\mathbf{R}_{0:N}$ is strongly regular if and only if all $\sigma_n \neq 0$ (equivalent to all $|k_n| \neq 1$); in such a case the number of its negative and positive eigenvalues is ν_N and $N - \nu_N$ where

$$\nu_N = n_-\{\mu_N, \cdots, \mu_1\} = \operatorname{Var}\{\sigma_N, \cdots, \sigma_0\}, \tag{52}$$

where n_- and Var stand, respectively, for the number of negative terms and the number of sign variations in the indicated sequences.

The magnitudes of the reflection coefficients $\{k_n\}$ of a Toeplitz matrix $\mathbf{T}_{0:N}$ also provide information about the location of the roots of the corresponding orthogonal polynomials $\{a_n(z)\}$ with respect to the unit circle. As is well known (see, e.g., [19]), $a_N(z)$ has all its roots strictly inside the unit circle if and only if $\mathbf{T}_{0:N}$ is *positive definite*, which by the foregoing discussion is equivalent to $\nabla_N > 0$. When $\mathbf{T}_{0:N}$ is indefinite, the magnitudes of $\{k_n\}$ determine the number of roots of $a_N(z)$ *inside* and *outside* the unit circle.

To be more specific, assume that we wish to locate the roots of $p(z)$, a given polynomial of degree N, with respect to the unit circle, and that $p(z)$ and $p^{\#}(z)$ do not have a common divisor, where $p^{\#}(z) := z^n[p(z^{-*})]^*$ denotes the conjugate reverse polynomial of $p(z)$. Then the Schur–Cohn test amounts to carrying out Levinson's recursion for Toeplitz matrices *in reverse order*, viz.,

$$za_{n-1}(z) = \frac{a_n(z) + k_n a_n^{\#}(z)}{1 - |k_n|^2}, \qquad k_n = -\frac{a_n(0)}{a_n^{\#}(0)} \tag{53a}$$

with the initialization $a_N(z) = p(z)$. This determines the reflection coefficients k_N, $k_{N-1}, \cdots, k_1$; the classical result of Cohn is that the number of roots of $p(z)$ inside (respectively, outside) the unit circle equals the number of positive (respectively, negative) P_n, $1 \leqq n \leqq N$, where

$$P_n := \prod_{i=n}^{N} (1 - |k_i|^2). \tag{53b}$$

If we use the balanced immittance-type recursions (10), then we shall have the coefficients $\{\delta_n\}$ or, equivalently, $\{\mu_n\}$ instead of the reflection coefficients $\{k_n\}$. The identity (47) implies that

$$\operatorname{sgn} P_n = \operatorname{sgn} \frac{\mu_{n-1}}{\mu_N} \quad \text{for } n = N, N-1, \cdots, 1 \tag{54a}$$

and consequently, the number of roots of $p(z)$ inside (respectively, outside) the unit circle equals the number of positive (respectively, negative) elements in the sequence

$$\left\{ \frac{\mu_{N-1}}{\mu_N}, \frac{\mu_{N-2}}{\mu_N}, \cdots, \frac{\mu_0}{\mu_N} \right\}. \tag{54b}$$

In particular, $p(z)$ is a stable polynomial (i.e., it has all its roots within the unit circle) if and only if all μ_n are positive (which will happen if and only if ∇_N is positive definite).

The balanced polynomials $f_n^B(z)$ are related to the orthogonal polynomials $a_n(z)$ of the Schur–Cohn test via the Toeplitz version of (15b), viz.,

$$f_n^B(z) = \psi_n^B[a_n(z) + \eta_n a_n^{\#}(z)],$$

where η_n and ψ_n^B have to satisfy the constraints (17) and (22), respectively. These constraints leave the parameters η_N, ψ_N^B, and ψ_{N-1}^B partially undetermined. Nevertheless, observe that $\{\mu_n\}$ are determined via (45b), viz.,

$$\mu_n = \frac{f_n^B(1)}{f_{n+1}^B(1)}, \qquad n = N, N-1, \cdots, 0 \tag{54c}$$

and that (54a)–(54c) hold *regardless* of the freedom in selecting the initialization.

The symmetric polynomials $f_n^B(z)$ are determined by propagating the balanced recursion (10) in reversed order, viz.,

$$zf_{n-1}^B(z) = (\delta_n z + \delta_n^*) f_n^B(z) - f_{n+1}^B(z), \qquad 1 \leqq n \leqq N-1, \tag{55a}$$

where

$$\delta_n = \left(\frac{f_{n+1}^B(0)}{f_n^B(0)} \right)^*. \tag{55b}$$

This recursion is initialized by $f_N^B(z)$ and $f_{N-1}^B(z)$ which, in turn, are determined by the parameters η_N, ψ_N^B, and ψ_{N-1}^B. Note that these three parameters determine all η_n for

$n < N$ (via (17a)), as well as all ψ_n^B for $n < N-1$ (via (22a)). The only constraint imposed on our initialization is (22b), i.e., $(\psi_N^B)^* = \eta_n \psi_N^B$, and $(\psi_{N-1}^B)^* = \eta_{N-1}\psi_{N-1}^B$. A particular choice that is consistent with this constraint is $\eta_N = 1$, $\psi_N^B = 1$, which results in

$$f_N^B(z) = p(z) + p^{\#}(z), \tag{56a}$$

where we used the fact that for Toeplitz matrices, $b_n(z) = a_n^{\#}(z)$, and where we set $a_N(z) = p(z)$, as in the Schur–Cohn procedure. Furthermore, we still maintain the property $|\eta_n| = 1$; in particular, $\eta_n = 1$ for matrices with real-valued elements.

Further simplification of the initial conditions for (55) may be achieved by a judicious choice of ψ_{N-1}^B, leading to a simplified expression for $f_{N-1}^B(z)$. An even simpler approach is to initialize (55) with $n = N$ rather than with $n = N-1$. This requires us to introduce the polynomial $f_{N+1}^B(z)$, which depends, via (55), on $f_N^B(z)$, $f_{N-1}^B(z)$, and the completely unconstrained parameter δ_N. The flexibility in selecting $f_{N-1}^B(z)$ and δ_N makes it possible to obtain a relatively simple expression for $f_{N+1}^B(z)$. Indeed, letting

$$\psi_{N-1}^B = 2\nu \frac{1-|k_N|^2}{1-k_N}, \qquad \delta_N = \nu \frac{1+k_N^*}{1-k_N^*} + \lambda,$$

where λ, ν are arbitrary positive constants, we find that

$$f_{N+1}^B(z) = q(z) + q^{\#}(z), \qquad q(z) := [\lambda(z+1) + \nu(z-1)]p(z). \tag{56b}$$

Moreover, these choices result in $\mu_N = (2\lambda)^{-1} > 0$, so that (54b) can be replaced by the sequence

$$\{\mu_{N-1}, \mu_{N-2}, \cdots, \mu_0\}. \tag{57a}$$

The number of negative elements in this sequence (i.e., the number of roots outside the unit circle) is also given by the number of sign changes in the sequence

$$\{f_N^B(1), f_{N-1}^B(1), \cdots, f_0^B(1)\}, \tag{57b}$$

which is always real-valued because: (i) $f_N^B(1) = p(1) + p^{\#}(1) = 2 \operatorname{Re} p(1)$ is real, (ii) the remaining $f_n^B(1)$ are obtained via $f_{n-1}^B(1) = \mu_{n-1} f_n^B(1)$, and (iii) μ_n are real.

Since μ_N (respectively, $f_{N+1}^B(1)$) does not appear in (57a) (respectively, (57b)), we can allow the limiting case $\lambda \to 0$ (with $\nu = 1$). This results in $q(z) = (z-1)p(z)$ so that

$$f_N^B(z) = p(z) + p^{\#}(z) = \frac{q(z) - q^{\#}(z)}{z-1}.$$

This is precisely the initialization that arises when the root-location procedure of Bistritz [1], [2] is applied to the *augmented polynomial* $q(z)$, which has the same root-distribution as $p(z)$ and an additional zero at $z = 1$. According to [1] and [2], the number of roots of $q(z)$ outside the unit circle equals the number of sign changes in the sequence

$$\{f_{N+1}^B(1), f_N^B(1), \cdots, f_0^B(1)\}.$$

Since the initial $f_{N+1}^B(1) = 0$ accounts for the zero at $z = 1$, we conclude that the remaining elements of this sequence determine the root-distribution of the polynomial $p(z)$. This coincides with our criterion (57b).

6. Concluding remarks. The Levinson and Schur algorithms showed how the Toeplitz (and quasi-Toeplitz) structure of linear equations could be used to provide an order of magnitude reduction in the amount of computation, from $O(N^3)$ to $O(N^2)$.

Normally we would not be too concerned with further reductions that do not affect the order of magnitude; however, the work of Bistritz [1]–[4] showed an alternative structure that achieves a reduction of exactly one-half in the number of multiplications. Such an improvement cannot be accidental, and that has been the motivation for the studies reported in this paper and our earlier paper [8], [9]. The first results of Bistritz (on stability tests) and of Delsarte and Genin (on the split Levinson algorithm) obtained this reduction in the amount of computation by carefully exploiting the persymmetry property of Toeplitz matrices. We were not completely satisfied with this approach because our earlier work on the Levinson algorithm showed that the algorithm could be generalized to close-to-Toeplitz matrices, and that this generalization was very simple for the class of (admissible) quasi-Toeplitz matrices, amounting essentially to a change in the initial conditions (see (2d)). Even though such non-Toeplitz matrices were not persymmetric and do not yield immittance symmetric polynomials, we were able to obtain a corresponding reduction in the number of multiplications, which seems to indicate that the persymmetry property does not fully explain the improved efficiency (this notwithstanding the fact that, at least in retrospect, the Levinson recursions for admissible quasi-Toeplitz matrices can be obtained from the usual Levinson algorithm by using the congruence (1c) and some algebraic manipulation).

The key to reduction in computation is really the proper use of two additional degrees of freedom that were always known but never fully exploited. These are:

(i) The possibility of linear transformations of the variables propagated in the Levinson and Schur algorithms, and especially the transformations (well known in circuit theory) between wave variables and immittance (voltage, current) variables;

(ii) The use of the three-term recursions (already noted in the classical work of Geronimus [18], [19]).

This is the approach developed in the present paper and in [8] and [9]. We may note that besides enabling a simple extension for Toeplitz to quasi-Toeplitz systems, our approach has also served to delimit the whole set of efficient Levinson and Schur algorithms.

Returning to the complexity reduction, we may remark that for us the main interest is not so much the reduction itself, which need not be significant in actual applications (e.g., studies of robustness and stability still need to be made), but more the reasons for the exact factor of two of reduction and the scope for its extension beyond the Toeplitz case. The simplicity of the two-step approach used in this paper showed that the same reduction in complexity could also be achieved for Hermitian quasi-Toeplitz matrices. How much further can they go? This is hard to say. However, our method of proof has recently enabled us to show that the reduction does *not* extend to *non-Hermitian* Toeplitz and quasi-Toeplitz matrices [21].

Appendix A. Derivation of general three-term recursions. It follows from (14b) that

$$\alpha_n(z)=\frac{\psi_n}{2\psi_{n-1}}\left\{(1-\eta_n k_n^*)z+\frac{\eta_n-k_n}{\eta_{n-1}}\right\}, \tag{A.1a}$$

$$\beta_n(z)=\frac{\psi_n}{2\psi_{n-1}}\left\{(1-\eta_n k_n^*)z-\frac{\eta_n-k_n}{\eta_{n-1}}\right\}, \tag{A.1b}$$

$$\gamma_n(z)=\frac{\psi_n}{2\psi_{n-1}}\left\{(1+\eta_n k_n^*)z-\frac{\eta_n+k_n}{\eta_{n-1}}\right\}, \tag{A.1c}$$

$$\delta_n(z)=\frac{\psi_n}{2\psi_{n-1}}\left\{(1+\eta_n k_n^*)z+\frac{\eta_n+k_n}{\eta_{n-1}}\right\}. \tag{A.1d}$$

Consequently,

$$\frac{\beta_{n+1}(z)}{\beta_n(z)} = \frac{\psi_{n+1}\psi_{n-1}\eta_{n-1}}{\psi_n^2\eta_n} \cdot \frac{\eta_n(1-\eta_{n+1}k_{n+1}^*)z-(\eta_{n+1}-k_{n+1})}{\eta_{n-1}(1-\eta_n k_n^*)z-(\eta_n-k_n)}.$$

This expression is independent of z if and only if

$$\frac{\eta_{n-1}(1-\eta_n k_n^*)}{\eta_n - k_n} = \mu$$

where μ is a constant independent of n. Therefore, $\{\eta_n\}$ are recursively determined via the recursion

$$\eta_n = \frac{\mu\eta_{n-1}+k_n}{1+k_n^*\mu\eta_{n-1}}, \tag{A.2}$$

which involves only two undetermined constants (μ and η_0). To be consistent with the real case, where $\eta_n = 1$ (see, e.g., [8], [9]), we choose $\eta_0 = 1$, $\mu = 1$, which results in the recursive relation (17). Note that we always get $|\eta_n| = 1$ with this choice of η_0, μ.

Incorporating the constraint (17) into the expressions (A.1) simplifies them to

$$\alpha_n(z) = \frac{\psi_n}{2\psi_{n-1}}(1-\eta_n k_n^*)(z+1), \tag{A.3a}$$

$$\beta_n(z) = \frac{\psi_n}{2\psi_{n-1}}(1-\eta_n k_n^*)(z-1), \tag{A.3b}$$

$$\gamma_n(z) = \frac{\psi_n}{2\psi_{n-1}}(1-\eta_n k_n^*)(\zeta_n z-\zeta_n^*), \tag{A.3c}$$

$$\delta_n(z) = \frac{\psi_n}{2\psi_{n-1}}(1-\eta_n k_n^*)(\zeta_n z+\zeta_n^*), \tag{A.3d}$$

where

$$\zeta_n := \frac{1+\eta_n k_n^*}{1-\eta_n k_n^*}. \tag{A.4}$$

The expressions (18a), (18b) for $f_{n+1}(z)$ and $g_n(z)$ are obtained by substituting (A.3) into (16a), (16c) and using the following easily established identities:

$$\frac{1+\zeta_n}{1+\zeta_n^*} = \frac{\eta_{n-1}}{\eta_n}, \tag{A.5a}$$

$$|\eta_n - k_n|^2(\zeta_n+\zeta_n^*) = 2(1-|k_n|^2). \tag{A.5b}$$

Appendix B. Properties of recursion coefficients. The constraint (22a), which characterizes the balanced recursion, implies that

$$\eta_{n+1}\frac{\psi_{n+1}^B}{(\psi_{n+1}^B)^*} = \eta_{n-1}\frac{\psi_{n-1}^B}{(\psi_{n-1}^B)^*}$$

where we have used (17) and the fact that $|\eta_n| = 1$. Therefore, (22b) follows for all n if we assume that it holds for $n = -1, 0$. Thus we must have

$$(\psi_{-1}^B)^* = \eta_{-1}\psi_{-1}^B, \qquad (\psi_0^B)^* = \psi_0.$$

It will be convenient, though by no means necessary, to have *the same initial conditions*

for all versions of the recursions. Taking this approach we conclude, via (27) and (29), that

$$\psi_0 = (1-k_0)^{-1} = (1-k_0^*)^{-1},$$

so that k_0 must be real and, consequently,

$$\eta_{-1} = \frac{1-k_0}{1-k_0^*} = 1.$$

This also proves that ψ_{-1} is real. In fact, (30) and (32) imply that

$$\psi_{-1} = \psi_0(1-|k_0|^2) = 1+k_0.$$

In summary, we can initialize all versions of the recursions with the same set of initial constants, viz.,

(B.1a) $$\psi_0 = (1-k_0)^{-1}, \qquad \psi_{-1} = 1+k_0$$

and

(B.1b) $$\eta_{-1} = 1 = \eta_0, \qquad \delta_0 = 1 = \lambda_0.$$

The only undetermined parameter is k_0, which can take any real value *except unity*. In particular, the initial conditions (21) are obtained by choosing $k_0 = -1$. Other simple choices of k_0, such as $k_0 = 0$, are also feasible.

Returning to establish the form (23) of the balanced recursions we denote the coefficient of $zf_n(z)$ in (18a) by δ_n and substitute into the latter (22a), (22b), viz.,

(B.2) $$\delta_n = \frac{\psi_{n+1}^B(\eta_{n+1}-k_{n+1})\eta_{n-1}}{\psi_n^B(\eta_n - k_n)\eta_n} = \frac{\psi_{n-1}^B\eta_{n-1}}{\psi_n^B(1-|k_n|^2)\eta_n} = \frac{(\psi_{n-1}^B)^*}{(\psi_n^B)^*(1-|k_n|^2)},$$

so that the coefficient of $f_n(z)$ in (23) is, indeed, $\delta_n z + \delta_n^*$. This also implies that

(B.3) $$\delta_n^* = \frac{\eta_n}{\eta_{n-1}}\delta_n$$

and, consequently, that

(B.4) $$\xi_n^* = \eta_n^{-1}\xi_n$$

where $\xi_n := \prod_{i=1}^n \delta_i^{-1}$, as in (33c). Similarly,

$$\lambda_n^* = (\delta_n^{-1}\delta_{n-1}^{-1})^* = \frac{\eta_{n-1}}{\eta_n}\frac{\eta_{n-2}}{\eta_{n-1}}\delta_n^{-1}\delta_{n-1}^{-1},$$

namely,

(B.5) $$\lambda_n^* = \frac{\eta_{n-2}}{\eta_n}\lambda_n.$$

The relations (33a), (33b) are established by a comparison of the leading coefficients in the polynomials $f_n^B(z)$, $f_n^M(z)$, etc. Since $f_0(z)$ is the same for all versions of the recursion and since $f_n^M(z)$ is monic, it follows, for instance, that

$$\frac{f_n^B(z)}{f_n^M(z)} = \prod_{i=0}^{n-1}\delta_i = \xi_{n-1}^{-1}$$

as well as

$$\frac{f_n^D(z)}{f_n^M(z)} = \prod_{i=1}^{n} \lambda_i^{-1} = \prod_{i=1}^{n} \delta_i \delta_{i-1} = \xi_n^{-1} \xi_{n-1}^{-1},$$

which implies that

$$\frac{f_n^D(z)}{f_n^B(z)} = \xi_n^{-1}.$$

The rest of the relations in (33) can be obtained in a similar manner.

In order to establish (47), we observe from (18b) that $\mu_{n-1} = f_{n-1}^B(1)/f_n^B(1)$ is given by

$$\mu_{n-1} = \frac{\psi_{n-1}^B}{\psi_n^B(1-\eta_n k_n^*)} = \psi_{n-1}^B \frac{\eta_{n-1}}{\psi_n^B(\eta_n - k_n)}$$

where the second equality invokes (17a). Consequently, we obtain, using (B.2),

$$\frac{\mu_{n-1}}{\mu_n} = \frac{\psi_{n-1}^B}{\psi_n^B} \cdot \frac{\psi_{n+1}^B \eta_{n-1}(\eta_{n+1} - k_{n+1})}{\psi_n^B \eta_n (\eta_n - k_n)} = \delta_n^*(1 - |k_n|^2) \cdot \delta_n$$

and, therefore,

$$\frac{\mu_m}{\mu_n} = \prod_{i=m+1}^{n} |\delta_i|^2 (1 - |k_i|^2). \tag{B.6}$$

This result does not depend on the choice of initialization.

REFERENCES

[1] Y. BISTRITZ, *Zero location with respect to the unit circle of discrete-time linear system polynomials*, Proc. IEEE, 72 (1984), pp. 1131–1142.

[2] ———, *A circular stability test for general polynomials*, Systems Control Lett., 7 (1986), pp. 89–97.

[3] ———, *Z-domain continued-fraction expansions for stable discrete systems polynomials*, IEEE Trans. Circuits and Systems, 32 (1985), pp. 1162–1166.

[4] ———, *A new unit circle stability criterion*, in Proc. 1983 Internat. Symposium on the Mathematical Theory of Networks and Systems, Beer-Sheva, Israel, 1983, pp. 69–87; Also in Lecture Notes in Control and Information Science, Vol. 58, Springer-Verlag, Berlin, New York, 1984, pp. 69–87.

[5] A. COHN, *Über die Anzahl der Wurzeln einer algebraischen Gleichung in einem Kreise*, Math. Z., 14 (1922), pp. 110–148.

[6] P. DELSARTE AND Y. GENIN, *The split Levinson algorithm*, IEEE Trans. Acoust. Speech Signal Process., 34 (1986), pp. 470–478.

[7] ———, *On the splitting of classical algorithms in linear prediction theory*, IEEE Trans. Acoust. Speech Signal Process., 35 (1987), pp. 645–653.

[8] Y. BISTRITZ, H. LEV-ARI, AND T. KAILATH, *Immittance-domain Levinson algorithms*, in Proc. 1986 IEEE Internat. Conference on Acoustics, Speech and Signal Processing, Tokyo, Japan, April 1986, pp. 253–256.

[9] ———, *Immittance-domain Levinson algorithms*, IEEE Trans. Inform. Theory, 35 (1989), pp. 675–682.

[10] P. DELSARTE, Y. GENIN, AND Y. KAMP, *Application of the index theory of pseudo-lossless functions to the Bistritz stability test*, Philips J. Res., 39 (1984), pp. 226–241.

[11] S. D. MORGERA AND H. KRISHNA, *The Szegö recurrence and the role of Hermitian and skew-Hermitian polynomial spaces*, manuscript.

[12] H. KRISHNA AND S. D. MORGERA, *The Levinson recurrence and fast algorithms for solving Toeplitz systems of linear equations*, IEEE Trans. Acoust. Speech Signal Process., 35 (1987), pp. 839–848.

[13] T. KAILATH, S.-Y. KUNG, AND M. MORF, *Displacement ranks of matrices and linear equations*, J. Math. Anal. Appl., 68 (1979), pp. 395–407; Also in Bull. Amer. Math. Soc., 1 (1979), pp. 769–773.

[14] H. LEV-ARI AND T. KAILATH, *Lattice-filter parameterization and modeling of nonstationary processes*, IEEE Trans. Inform. Theory, 30 (1984), pp. 2–16.
[15] J. D. MARKEL AND A. H. GRAY, JR., *Linear Prediction of Speech*, Springer-Verlag, New York, 1978.
[16] T. KAILATH, *A theorem of I. Schur and its impact on modern signal processing*, in I. Schur Methods in Operator Theory and Signal Processing, Operator Theory: Advances and Applications, Vol. 18, I. Gohberg, ed., Birkhäuser-Verlag, Basel, Switzerland, 1986, pp. 9–30.
[17] H. W. BODE, *Network Analysis and Feedback Amplifier Design*, Van Nostrand, New York, 1945.
[18] YA. L. GERONIMUS, *Polynomials orthogonal on a circle and their applications*, Amer. Math. Soc. Trans., No. 104, 1954 (Russian Publication, 1948).
[19] ———, *Orthogonal Polynomials*, Consultant's Bureau, New York, 1961.
[20] P. DELSARTE AND Y. GENIN, *Multichannel singular predictor polynomials*, IEEE Trans. Circuits and Systems, 35 (1988), pp. 190–200.
[21] Y. BISTRITZ, H. LEV-ARI, AND T. KAILATH, *Immittance versus scattering domains fast algorithms for non-Hermitian Toeplitz and quasi-Toeplitz matrices*, Linear Algebra Appl., 124 (1989), pp. 847–888.
[22] Y. BISTRITZ AND T. KAILATH, *Inversion and factorization of non-Hermitian quasi-Toeplitz matrices*, Linear Algebra Appl., 98 (1988), pp. 77–121.
[23] K. P. BUBE AND R. BURRIDGE, *The one-dimensional inverse problem of reflection seismology*, SIAM Rev., 25 (1983), pp. 497–559.