SIAM J. SCI. STAT. COMPUT.
Vol. 7, No. 1, January 1986

009

COMPUTING THE MINIMUM EIGENVALUE OF A SYMMETRIC POSITIVE DEFINITE TOEPLITZ MATRIX*

GEORGE CYBENKO† AND CHARLES VAN LOAN‡

Abstract. A method for computing the smallest eigenvalue of a symmetric positive definite Toeplitz matrix is given. It relies solely upon the Levinson–Durbin algorithm. The procedure involves a combination of bisection and Newton's method. Good starting values are also shown to be obtainable from the Levinson–Durbin algorithm.

Key words. Toeplitz, eigenvalue, signal processing

1. Introduction. Recent progress in signal processing and estimation has generated considerable interest in the problem of computing the minimal eigenvalue of a Toeplitz matrix. The fundamental modeling and solution that led to this are due to Pisarenko [P73], while more recently numerous authors have discussed the computational aspects of the problem [F83], [H80], [H83a], [H83c].

In this paper we shall not discuss the underlying assumptions, merits, or potential applications of the model—instead pointing the interested reader to the literature concerned with these issues [H83a], [P73]. We hasten to add that the quantities of ultimate interest in applications are the roots of the polynomial whose coefficients are given by the eigenvector associated with the minimal eigenvalue of the Toeplitz matrix. We shall only discuss the computation of the minimal eigenvalue noting that the associated eigenvector can be obtained as a by-product. Furthermore, methods exist for computing the roots that altogether avoid the explicit formation of the eigenvector [C84b].

The essence of our minimum eigenvalue procedure involves solving systems of shifted Yule–Walker (YW) systems. Initially, the solutions to these systems are used in a bisection scheme that repeatedly halves a bracketing subinterval. Subsequently, a Newton iteration takes over that quadratically converges to the desired eigenvalue. We stress the fact that only YW systems are involved—an important point since extremely efficient methods for YW systems exist. (They require half the computational resources needed by general symmetric Toeplitz system solvers.)

In an absolute sense, only modest use is made of Toeplitz structure. Indeed, this is true of all currently known Toeplitz eigenvalue solvers. The study of the eigenstructure of finite Toeplitz matrices is proceeding rather slowly. Recent developments include [C84a], [C84b], [D83]. An indication of the collective ignorance about Toeplitz eigenstructure is that the inverse eigenvalue problem for real symmetric Toeplitz matrices is currently unsolved. We suspect that the process of designing efficient algorithms for this problem will go hand in hand with the uncovering of Toeplitz eigenstructure properties.

Our paper is organized as follows. Section 2 describes a rational function intimately related to the eigenvalue problem for Hermitian matrices. Section 3 specializes the

* Received by the editors May 9, 1984.

† Department of Mathematics, Tufts University, Medford, Massachusetts 02155 and the Statistics Center, Massachusetts Institute of Technology, Cambridge, Massachusetts 02135. The work of this author was supported in part by the National Science Foundation under contract NSF MCS 8003364.

‡ Department of Computer Science, Cornell University, Ithaca, New York 14853. The work of this author was supported in part by the National Science Foundation under contract NSF MCS 8004106, and by the Office of Naval Research under contract ONR N00014-83-K-0640.

discussion to real symmetric Toeplitz matrices and then develops our bisection/Newton scheme. In the last section we discuss the numerical behavior of our procedure.

2. A rational eigenvalue equation. In this section we derive a rational function from a given Hermitian matrix that has the property that its zeros are eigenvalues of the original matrix. A feature of this rational function is that both it and its derivatives are easily evaluated thereby making Newton-type schemes feasible. Strictly speaking, parts of our derivation are not new and can be found in [W65], but we present the details for the sake of completeness.

Let T be an $n \times n$ Hermitian matrix partitioned as follows:

$$T = \begin{bmatrix} \gamma & r^* \\ r & G \end{bmatrix} \begin{matrix} 1 \\ n-1 \end{matrix} . \qquad \begin{matrix} 1 & n-1 \end{matrix}$$

Here, r^* denotes the conjugate transpose of r. It is well known [G83] that the eigenvalues of T and G are real and satisfy an interlacing property. In particular, if $\lambda_i(T)$ and $\lambda_i(G)$ are the ith largest eigenvalues of T and G respectively then

$$\lambda_n(T) \leqq \lambda_{n-1}(G) \leqq \lambda_{n-1}(T) \leqq \cdots \leqq \lambda_2(T) \leqq \lambda_1(G) \leqq \lambda_1(T).$$

Note that if T has a repeated eigenvalue, then the repeated value is also an eigenvalue of G. Adopting the notation

$$\lambda_{\min} \equiv \lambda_n(T),$$

we shall assume throughout this paper that

$$d = \lambda_{n-1}(G) = \lambda_{\min} > 0. \tag{2.1}$$

The strict separation of $\lambda_{n-1}(G)$ and $\lambda_{\min}$ guarantees that the eigenvector associated with $\lambda_{\min}$ is unique up to scalar premultiplication. It is a realistic assumption in many important problems such as the estimation of Pisarenko frequencies. See [C84b].

Suppose

$$\begin{bmatrix} \gamma & r^* \\ r & G \end{bmatrix} \begin{bmatrix} \alpha \\ y \end{bmatrix} = \lambda_{\min} \begin{bmatrix} \alpha \\ y \end{bmatrix} \tag{2.2}$$

where it is assumed that α and y are not both zero. From this equation we obtain

$$\begin{aligned} \gamma\alpha + r^* y &= \lambda_{\min}\alpha, \\ \alpha r + G y &= \lambda_{\min} y. \end{aligned} \tag{2.3}$$

We must have $\alpha \neq 0$ for otherwise $Gy = y_{\min} y$ contradicting (2.1). Noting that $G - \lambda_{\min} I$ is positive definite we obtain the following rational equation for $\lambda_{\min}$:

$$\gamma - r^*(G - \lambda_{\min} I)^{-1} r - \lambda_{\min} = 0. \tag{2.4}$$

Thus, the smallest eigenvalue of T is the smallest root of the rational function

$$f(\lambda) = \gamma - \lambda - r^*(G - \lambda I)^{-1} r. \tag{2.5}$$

In addition, $f(\lambda)$ has the following important properties if $0 \leqq \lambda < \lambda_{n-1}(G)$:

$$f'(\lambda) = -1 - \|(G - \lambda I)^{-1} r\|_2^2 \leqq -1, \tag{2.6}$$

$$f''(\lambda) = -2r^*(G - \lambda I)^{-3} r \leqq 0. \tag{2.7}$$

Now consider the following Newton iteration:

ALGORITHM 2.1. Let $\lambda \in [\lambda_{\min}, \lambda_{n-1}(G))$ be given along with a tolerance $\delta > 0$.
Do Until $(|f(\lambda)| < \delta/(1+\|w\|_2^2)^{1/2})$

Solve $(G-\lambda I)w = -r$ for w.

$$\lambda := \lambda + \frac{\gamma + r^* w - \lambda}{1 + w^* w} = \lambda - \frac{f(\lambda)}{f'(\lambda)}.$$

Properties (2.6) and (2.7) ensure that the iteration converges to $\lambda_{\min}$. To see this assume that $\lambda \in (\lambda_{\min}, \lambda_{n-1}(G))$ and set

$$\lambda_+ = \lambda - \frac{f(\lambda)}{f'(\lambda)}.$$

Since f is monotone decreasing in this interval, it follows that both $f(\lambda)$ and $f'(\lambda)$ are negative. Thus, $\lambda_+ < \lambda$. On the other hand, from truncated Taylor series we have

$$0 = f(\lambda_{\min}) = f(\lambda) + f'(\lambda)(\lambda_{\min} - \lambda) + \frac{f''(\zeta)}{2}(\lambda_{\min} - \lambda)^2$$

with $\zeta \in [\lambda_{\min}, \lambda]$. It follows that

$$\lambda_+ - \lambda_{\min} = \frac{f''(\zeta)}{2f'(\lambda)}(\lambda_{\min} - \lambda)^2 > 0. \tag{2.8}$$

Thus, the iterates in the algorithm converge monotonically to $\lambda_{\min}$ from the right and at a rate that is ultimately quadratic. Note from (2.8) that in the limit we have

$$(\text{error in new } \lambda) \cong C \cdot (\text{error in old } \lambda)^2$$

where

$$C = \frac{f''(\lambda_{\min})}{2f'(\lambda_{\min})} = \frac{w^*(G - \lambda_{\min} I)^{-1} w}{1 + w^* w}$$

and $w = -(G - \lambda_{\min} I)^{-1} r$. Since $\|(G - \lambda_{\min} I)^{-1}\|_2 = 1/d$ it is easy to show that $C \leqq 1/d$. It follows that Algorithm 2.1 may converge slowly in problems where the separation d is small. We return to this point later.

The termination criteria in Algorithm 2.1 gives good absolute error in the final λ provided the tolerance δ is small enough. This follows from

$$\left\| \begin{bmatrix} \gamma & r^* \\ r & G \end{bmatrix} \begin{bmatrix} 1 \\ w \end{bmatrix} - \lambda \begin{bmatrix} 1 \\ w \end{bmatrix} \right\|_2 = \left\| \begin{bmatrix} f(\lambda) \\ 0 \end{bmatrix} \right\|_2 = |f(\lambda)|.$$

Applying standard Hermitian matrix perturbation theory (see [G83]), we may conclude that there exists an exact eigenvalue λ_e of T that satisfies

$$|\lambda - \lambda_e| < |f(\lambda)|(1 + \|w\|_2^2)^{1/2} < \delta.$$

If δ is sufficiently small compared to the separation d, then one can ensure that $\lambda_e = \lambda_{\min}$.

Despite the nice mathematical properties of Algorithm 2.1, its practical value hinges on two critical factors: how is the starting value determined and how is the linear system $(G - \lambda I)w = -r$ to be solved? We address these questions in the next section for the case when T is symmetric, positive definite, and Toeplitz.

3. The symmetric positive definite Toeplitz case. Let $(t_0, t_1, \cdots, t_{n-1})$ be the first row of a symmetric positive definite Toeplitz matrix $T = (t_{ij})$, i.e., $t_{ij} = t_{|i-j|}$. Assume that T is normalized so that $t_0 = 1$ and partition it as follows:

$$T = \begin{bmatrix} 1 & r^T \\ r & G \end{bmatrix}, \qquad r^T = (t_1, \cdots, t_{n-1}).$$

Recall that in order to apply Algorithm 2.1 we must find a starting value λ that belongs to the interval $[\lambda_n(T), \lambda_{n-1}(G))$. This requirement can be couched in the language of signatures. The signature sig (A) of a symmetric matrix A is a triplet of integers (neg, z, pos) where neg, z, and pos are the number of negative, zero and positive eigenvalues of A. Our starting value problem is to find λ such that sig $(G - \lambda I) = (0, 0, n-1)$ while sig $(T - \lambda I) = (1, 0, n-1)$ or $(0, 1, n-1)$.

This problem can be solved by exploiting the well-known Levinson–Durbin algorithm:

ALGORITHM 3.1.
$E_0 = 1$
For $i = 1$ to $n-1$
$$k_i = -\left(t_i + \sum_{j=1}^{i-1} a_{i-1,j} t_j\right) \Big/ E_{i-1}$$
For $j = 1$ to $i - 1$
$$a_{ij} = a_{i-1,j} + k_i a_{i-1,i-j}$$
$a_{ii} = k_i$
$E_i = E_{i-1}(1 - k_i^2)$

The a_{ij} satisfy the Yule–Walker (YW) systems

$$\begin{bmatrix} 1 & t_1 & \cdots & t_{i-1} \\ t_1 & 1 & & \vdots \\ \vdots & & \ddots & \vdots \\ t_{i-1} & & \cdots & 1 \end{bmatrix} \begin{bmatrix} a_{i1} \\ \vdots \\ a_{ii} \end{bmatrix} = -\begin{bmatrix} t_1 \\ \vdots \\ t_i \end{bmatrix}$$

for $i = 1, \cdots, n-1$. The quantities k_i and E_i are referred to as the ith partial correlation coefficient and the ith prediction error respectively. (k_i is also known as the ith reflection coefficient.) See [G83] for a discussion of Algorithm 3.1.

In [C80] it is shown that if

$$L = \begin{bmatrix} 1 & & & & \\ a_{11} & 1 & & 0 & \\ a_{22} & a_{21} & 1 & & \\ \vdots & & & \ddots & \\ a_{n-1,n-1} & & & \cdots & 1 \end{bmatrix}$$

then

$$LTL^T = \operatorname{diag}(1, E_1, \cdots, E_{n-1}). \tag{3.1}$$

Since signature is preserved under congruence transformations by the Sylvester Law of Inertia, all of the E_i are positive since T is positive definite.

However, if we apply Algorithm 3.1 to the normalized Toeplitz matrix $(T - \lambda I)/(1-\lambda)$ and if the algorithm runs to completion, then the number of negative E_i that are generated equals the number of eigenvalues of $T - \lambda I$ that are negative, i.e., the number of T's eigenvalues that are strictly less than λ. The caveat "runs to completion" must be added because it is possible for one of the E_i to be zero if Algorithm 3.1 is applied to an indefinite T. Adapting the algorithm so that computes

(3.1) for $T := (T-\lambda I)/(1-\lambda)$ gives

ALGORITHM 3.2.

$i=0$

$E_0=1$

Do While $(E_i>0 \ \& \ i<n-1)$

$i:=i+1$

$$k_i = -\left(t_i + \sum_{j=1}^{i-1} a_j t_j\right) \Big/ [(1-\lambda)E_{i-1}]$$

$$\begin{bmatrix} a_1 \\ \vdots \\ a_{i-1} \\ a_i \end{bmatrix} := \begin{bmatrix} a_1 \\ \vdots \\ a_{i-1} \\ 0 \end{bmatrix} + k_i \begin{bmatrix} a_{i-1} \\ \vdots \\ a_1 \\ 1 \end{bmatrix}$$

$$E_i = E_{i-1}(1-k_i^2)$$

We have dropped the double subscripting of the a's since we need only be in possession of the most recent YW solution at any one time.

Note that if the loop terminates because $i=n-1$, then we have

$$(G-\lambda I)\begin{bmatrix} a_1 \\ \vdots \\ a_{n-1} \end{bmatrix} = -\begin{bmatrix} t_1 \\ \vdots \\ t_{n-1} \end{bmatrix}.$$

Recall that being able to solve this shifted YW system is critical to Algorithm 2.1, the Newton iteration for $f(\lambda)$.

Equally important, the final value of i in Algorithm 3.2 enables us to determine the position of λ with respect to $\lambda_{\min}$ and $\lambda_{n-1}(G)$:

(a) If $i=n-1$ and $E_{n-1}>0$, then $\lambda<\lambda_{\min}$.

(b) If $i=n-1$ and $E_{n-1}\leqq 0$, then $\lambda_{\min}\leqq\lambda<\lambda_{n-1}(G)$.

(c) If $i<n-1$ then $\lambda_{n-1}(G)\leqq\lambda$.

Hence, Algorithm 3.2 can be used in a bisection scheme to position λ eventually in the interval $[\lambda_{\min}, \lambda_{n-1}(G))$. Thereafter, it can be used to carry out the Newton iteration. All we need is an initial interval $[\alpha, \beta]$ with the property that

$$\alpha \leqq \lambda_{\min} \leqq \beta. \tag{3.2}$$

ALGORITHM 3.3.

Compute α and β satisfying (3.2) and let $\delta>0$ be a given tolerance.

$k=0$

$\lambda^{(0)}=(\alpha+\beta)/2$

Do Until $(|\lambda^{(k)}-\lambda^{(k-1)}|\leqq\delta|\lambda^{(k-1)}|)$

Apply Algorithm 3.2 with $\lambda=\lambda^{(k)}$ to generate i and $a_1, \cdots, a_i$.

$k:=k+1$

If $(i<n-1)$

then

$\beta=\lambda$; $\lambda^{(k)}=(\alpha+\beta)/2$

else

If $(E_{n-1}>0)$

then

$\alpha=\lambda$; $\lambda^{(k)}=(\alpha+\beta)/2$

else

$$\lambda^{(k)}=\lambda+\frac{1-\lambda+t_1a_1+\cdots+t_{n-1}a_{n-1}}{1+a_1^2+\cdots+a_{n-1}^2}$$

The last expression for $\lambda^{(k)}$ above is the same as the λ update expression in Algorithm 2.1 with $\gamma = 1$, $r^T = (t_1, \cdots, t_{n-1})^T$ and $w^T = (a_1, \cdots, a_{n-1})^T$.

There are several possible ways to choose the initial bracketing interval $[\alpha, \beta]$. For example, we could set $[\alpha, \beta] = [0, 1]$. The choice for β follows from the inequality $\lambda_{\min} \leqq e_1^T T e_1 = 1$ where $e_1 = (1, 0, \cdots, 0)^T$. There are, however, more refined ways to get the initial interval:

Method 1. See $[\alpha, \beta] = [0, 1 - |t_1|]$. Since the smallest eigenvalue of

$$T_1 = \begin{bmatrix} 1 & t_1 \\ t_1 & 1 \end{bmatrix}$$

is given by $1 - |t_1|$, we have from separation theory that $1 - |t_1| = \lambda_2(T_1) \geqq \lambda_n(T)$.

Method 2. Set $[\alpha, \beta] = [0, \min_i \{1 - |t_i|\}]$. The reasoning is the same as for Method 1 with t_1 replaced by t_i. Note that $\left[\begin{smallmatrix} 1 & t_i \\ t_i & 1 \end{smallmatrix}\right]$ is a principal submatrix of T.

Method 3. Set $[\alpha, \beta] = [0, E_{n-2}(1 - |k_{n-1}|)]$ where E_{n-2} and k_{n-1} are generated by Algorithm 3.3 with λ_0. To understand the choice for β, consider the effect of replacing t_{n-1} with $\tilde{t}_{n-1} = t_{n-1} + \varepsilon$ in Algorithm 3.3 and that we set $\lambda = 0$. Nothing changes except during the last pass through the loop when we compute

$$\tilde{k}_{n-1} = -\left(\tilde{t}_{n-1} + \sum_{j=1}^{n-2} a_j t_j\right) \Big/ E_{n-2} = -\frac{\varepsilon}{E_{n-2}} + k_{n-1}.$$

Note that if we choose ε so that $1 - \tilde{k}_{n-1}^2 = 0$, then the resulting $\tilde{E}_{n-1}$ will be zero. Thus, a perturbation of size ε transforms T into a singular matrix. It follows that $\lambda_n(T) \leqq \varepsilon$. The choice for β is the smaller of the two ε values that render $\tilde{k}_{n-1}^2 = 1$.

Method 4. Set

$$[\alpha, \beta] = \left[\frac{1}{\sqrt{n}\|T^{-1}\|_\infty}, \frac{\sqrt{n-1}}{\|G^{-1}\|_\infty}\right].$$

The value for α follows from the inequality

$$\frac{1}{\lambda_n(T)} = \|T^{-1}\|_2 \leqq \sqrt{n}\|T^{-1}\|_\infty.$$

Here, $\|\cdot\|_\infty$ denotes maximum row sum. The value for β follows from

$$\frac{1}{\lambda_n(T)} > \frac{1}{\lambda_{n-1}(G)} = \|G^{-1}\|_2 \geqq \sqrt{n-1}\|G^{-1}\|_\infty.$$

The quantities $\|T^{-1}\|_\infty$ and $\|G^{-1}\|_\infty$ can be calculated in $O(n^2)$ operations and $O(n)$ storage using the Trench algorithm [T64].

4. Analysis, discussion and numerical experiments. Another method for finding $\lambda_{\min}$ via the Levinson–Durbin algorithm is presented in [H80]. They propose solving $f(\lambda) = 0$ ($E_n(\mu) = 0$ in their notation) using a linear interpolation scheme. A key aspect of our work and what distinguishes it from [H80] is the recognition that one can apply Newton's method using by-products from the Levinson–Durbin algorithm. In addition, we have attempted to handle the problem of starting values more rigorously than [H80].

The importance of only having to solve Yule–Walker systems should be stressed. Methods based on inverse iteration, for example, require the solution of general Toeplitz systems. This doubles the amount of work per step. Moreover, there currently exist highly concurrent algorithms and VLSI architectures for solving Yule–Walker systems in $O(n)$ time. See [K83].

The total amount of work required by Algorithm 3.3 is determined by the number of YW systems that must be solved. The number N_B of bisection steps is bounded above by

$$N_B \leqq -\log_2 [d/(\beta-\alpha)]+1.$$

Note that during this phase of the algorithm, calls to Algorithm 3.2 do not require a full $n-1$ steps so it is a little hard to quantify the overall work. As a function of n, N_B appears to grow as $\log(n)$. A simple explanation of this is possible if we assume that the eigenvalues of T are uniformly distributed. In this case, the distance $\lambda_{n-1}(G)-\lambda_{\min}$ is roughly $1/n^2$. Hence the worst case limit on N_B is proportional to $\log(n)$.

The number of Newton steps N_N tends to be around 5 or 6 based on our experience with numerous examples a subset of which we now describe. For each $n = 11, 21, \cdots, 91$ we generated 25 random positive definite symmetric Toeplitz matrices. These matrices had the form

$$T = m \sum_{k=1}^{n} w_k T_{2\pi\theta_k}$$

where n is the dimension, m is chosen so that T is normalized,

$$T_\theta = (t_{ij}) = (\cos(\theta(i-j))),$$

and the w_k and θ_k are uniformly distributed random numbers taken from [0, 1]. It can be shown that T_θ is rank two, symmetric, semidefinite, and Toeplitz.

Table 1 summarizes the results of these experiments. Only initial interval Methods 1 and 3 were tabulated. Methods 2 and 4 were too similar in performance to Methods 1 and 3 for us to report. (Note: in recording the work associated with Method 3 the single call to Algorithm 3.2 required to compute β is accounted for in the table.)

TABLE 1
Behavior of Algorithm 3.3 ($\delta = 10^{-6}$) *based on* 25 *random examples per dimension.*

	Starting values via Method 1		Starting values via Method 3	
Order	Bisection steps	Newton steps	Bisection steps	Newton steps
11	4.8	5.0	5.7	4.8
21	8.5	5.7	4.8	5.4
31	9.7	5.3	6.6	5.0
41	10.0	4.6	6.7	5.2
51	11.7	5.8	8.1	5.5
61	12.1	5.0	9.2	5.3
71	12.0	5.3	9.2	5.2
81	13.6	5.4	9.8	5.0
91	11.6	5.0	8.4	5.7

The matrices were generated by a Fortran program and the eigenvalues $\lambda_{\min}$ and $\lambda_{n-1}(G)$ were computed by the EISPACK routine RS [S76]. Although our generation technique was guaranteed to generate at least a semi-definite matrix (definite with probability one) rounding errors led to a generation of some isolated slightly indefinite cases. Although indefinite matrices (due to finite arithmetic) ought to be expected in practice, they provide no realistic test for our procedure. In fact, if we know that our data has t significant bits (floating point), then more than t calls to the bisection step is useless. Given the quality of our data, after t steps of bisection we must conclude that the matrix is either not definite or that the condition $|\lambda_{n-1}(G)-\lambda_{\min}|<2^{-t}$ holds and so to the precision of our data, $\lambda_n(G)=\lambda_{\min}$. This follows from standard eigenvalue perturbation arguments [P80].

While on the subject of small separations, it is interesting to point out that $1/d$ measures the sensitivity or "condition" of $\lambda_{\min}$'s eigenvector $x_{\min}$. If this quantity is large then small perturbations in T can induce large changes in $x_{\min}$. (See [G83, p. 271].) As we mentioned in the introduction, the computation of $\lambda_{\min}$ is frequently just the first step in computing $x_{\min}$, the "real" quantity of interest. Thus, slow convergence in Algorithm 3.3 goes hand in hand with ill-conditioning in the underlying $x_{\min}$ problem.

The actual procedure was implemented in C on a DEC-10. Computations were done in the "double" data type. The tolerance δ in Algorithm 3.3 was set to 10^{-6}. The Newton iteration terminated successfully on all strictly separated trials and gave six significant digit agreement with EISPACK generated solutions.

Finally, we recommend Method 3 among the various procedures that we described for obtaining an initial interval. However, it is conceivable that the simplicity of Method 1 might make it more appealing than Method 3 in real time processing situations with elementary processors.

Acknowledgment. We are each indebted to the useful recommendations of the referees.

REFERENCES

[B80] R. BITMEAD AND B. D. O. ANDERSON, *Asymptotically fast solutions of Toeplitz and related systems of equations*, Lin. Alg. Appl., 34 (1980), pp. 103–116.

[B82] R. BRENT AND F. LUK, *A systolic array for linear-time solution of Toeplitz systems of equations*, Cornell Computer Science Technical Report TR82-526, Ithaca, NY, 1982.

[C80] G. CYBENKO, *The numerical stability of the Levinson–Durbin algorithm for Toeplitz systems of equations*, this Journal, 1 (1980), pp. 303–319.

[C84a] ———, *On the eigenstructure of Toeplitz matrices*, IEEE Trans. Acoust., Speech, SIgnal Proc., ASSP, 31 (1984), pp. 918–920.

[C84b] ———, *Computing Pisarenko frequency estimates*, Proc. 1984 Princeton Conference on Information Systems and Sciences, March 14–16, 1984, pp. 587–591.

[D82] T. S. DURRANI AND K. C. SHARMAN, *Excitation of an eigenvector oriented spectrum from the MESA coefficients*, IEEE Trans. Acoust., Speech, Signal Proc., ASSP, 30 (1982), pp. 649–651.

[D83] P. DELSARTE AND Y. GENIN, *Spectral properties of finite Toeplitz matrices*, Proc. 1983 Mathematical Theory of Networks and Systems, Beer-Sheva, Israel, 1983.

[F83] D. FUHRMAN AND BEDE LIU, *Approximating the eigenvectors of a symmetric Toeplitz matrix*, 1983, preprint.

[G83] G. H. GOLUB AND C. VAN LOAN, *Matrix Computations*, Johns Hopkins Univ. Press, Baltimore, 1983.

[H80] Y. H. HU AND S. Y. KUNG, *Computation of the minimum eigenvalue of a Toeplitz matrix by the Levinson algorithm*, Proceedings SPIE 25th International Conference (Real Time Signal Processing), San Diego, August 1980, pp. 40–45.

[H83a] ———, *A Toeplitz eigensystem solver*, 1983, preprint.

[H83b] Y. H. HU, *Resolution enhanced spectrum estimation via Toeplitz eigensystem solvers*, Proceedings 22nd IEEE Conference on Decision and Control, San Antonio, TX, Dec. 14–16, 1983, pp. 1345–1346.

[H83c] Y. H. HU AND S. Y. KUNG, *Highly concurrent Toeplitz eigensystem solver for high resolution spectral estimation*, Proc. IEEE 1983 ICASSP, Boston, 1983, pp. 1422–1425.

[K83] S. Y. KUNG AND Y. H. HU, *A highly concurrent algorithm and architecture for solving Toeplitz systems*, IEEE Trans. Acoust., Speech, Signal Proc., ASSP-31 (1983), pp. 66–76.

[P73] V. F. PISARENKO, *The retrieval of harmonics from a covariance function*, Geophys. J. Royal Astron. Soc., 33 (1973), pp. 347–366.

[P80] B. N. PARLETT, *The Symmetric Eigenvalue Problem*, Prentice-Hall, Englewood-Cliffs, NJ.

[S76] B. T. SMITH et al., *Matrix Eigensystem Routines*—EISPACK *Guide*, Springer-Verlag, New York, 1976.

[T64] W. J. TRENCH, *An algorithm for the inversion of finite Toeplitz matrices*, J. Soc. Ind. Appl. Math., 12 (1964), pp. 515–522.

[W65] J. WILKINSON, *The Algebraic Eigenvalue Problem*, Oxford, New York, 1965.