Structure and Arithmetic Complexity of Products and Inverses of Toeplitz Matrices

Elliot Linzer*
1312 S. W. Mudd Building
Department of Electrical Engineering
Columbia University
New York, New York 10027

and

Martin Vetterli‡
1312 S. W. Mudd Building
Department of Electrical Engineering
and Center for Telecommunications Research
Columbia University
New York, New York 10027

Submitted by Richard A. Brualdi

ABSTRACT

We derive a formula for the product of two Toeplitz matrices that is similar to the Trench formula for the inverse of a Toeplitz matrix. We then derive upper and lower bounds for number of multiplications required to compute the inverse or the product of Toeplitz matrices and consider several special cases, e.g., symmetry, as well. The lower bounds for the general cases are in agreement with earlier results, but the specialized lower bounds and all the upper bounds are new. Both upper and lower bounds are $O(n^2)$ and differ only in lower order terms.

*Work supported in part by a National Science Foundation Graduate Fellowship and carried out in partial fulfillment of the requirements for the Ph.D. degree in the Department of Electrical Engineering at Columbia University.

‡Work supported in part by the National Science Foundation under grants CDR-84-21402 and MIP-88-08277.

655 Avenue of the Americas, New York, NY 10010 0024-3795/91/$3.50

1. INTRODUCTION

An $(n + 1) \times (n + 1)$ Toeplitz matrix is a matrix of the form

$$A = \begin{pmatrix} a_0 & a_1 & \cdots & a_{n-1} & a_n \\ a_{-1} & a_0 & \cdots & a_{n-2} & a_{n-1} \\ \vdots & \vdots & & \vdots & \vdots \\ a_{1-n} & a_{2-n} & \cdots & a_0 & a_1 \\ a_{-n} & a_{1-n} & \cdots & a_{-1} & a_0 \end{pmatrix}, \tag{1}$$

i.e., $A_{i+1,j+1} = A_{i,j}$. Toeplitz matrices play an important role in many problems in system theory and signal processing [3, 4]. A similar and important class of matrices are Hankel matrices. A matrix H is a Hankel matrix if $H_{i-1,j+1} = H_{i,j}$. In this paper, we will deal only with Toeplitz matrices. Because Hankel matrices are related to Toeplitz matrices through a simple permutation, it is a trivial matter to apply the results of this paper to Hankel matrices.

The solution of linear systems of equations involving a Toeplitz matrix is a very common problem in practice. Fast, $O(n^2)$ [3, 9, 14], and even superfast, $O(n \log^2 n)$ [1, 2], algorithms exist for this problem. To compute the complete inverse of a Toeplitz matrix, one can use the Trench formula [14, 3], which relates the border of the inverse of a Toeplitz matrix to its interior through the sum of two outer products. The resulting algorithm uses $O(n^2)$ operations, as compared with $O(n^3)$ for the more general Gaussian elimination.

The product of Toeplitz matrices has to be formed for certain problems in spectral estimation [18] and the solution of banded Toeplitz systems [10]. We derive an equivalent of the Trench formula for the product of two Toeplitz matrices. With this new formula we are able to show that the computation of the product of two Toeplitz matrices also requires at most $O(n^2)$ operations.

The question arises whether these $O(n^2)$ methods for solving inverse and product problems are close to what an optimal algorithm could achieve. We present a simple analysis based on the relationship that exists between outer products and Toeplitz products or inverses to show that the lower bound is indeed $O(n^2)$. It turns out that the lower bounds for general Toeplitz matrices have been derived with a different method by Makarov [11]. Our proof is not only simpler, but its structure allows us to extend it easily to special cases, such as when the matrices are symmetric or for the squaring problem, where we show new lower bounds.

Relating Toeplitz products or inverses to sums of outer products reduces much of the problem of finding good algorithms for these problems to the simpler problem of finding good algorithms for computing sums of outer products. We use this approach to derive efficient algorithms for Toeplitz inversion and multiplication. These algorithms give us $O(n^2)$ upper bounds for the number of multiplications which differ from the lower bounds only in lower order terms.

In the next section, we introduce some notation and indicate the lower bounding techniques that will be used. Section 3 relates the inverse and product of Toeplitz matrices to some structured outer product problems. Section 4 investigates the complexity of these outer products. Sections 2–4 are used in Sections 5 and 6 to come up with complexity bounds as well as practical algorithms for the product and inverse problems.

2. BOUNDING TECHNIQUES

In this paper, we are concerned with finding upper and lower bounds for the number of multiplications required to perform various computations on Toeplitz matrices. The emphasis is on multiplications because a theoretical framework exists which allows us to easily find lower bounds for the number of multiplications required for a broad range of computations.

Finding an upper bound for the number of multiplications required to perform a given computation is straightforward: we describe an algorithm and count the number of multiplications used. In all of the algorithms described in this paper, the number of additions used will be of the same order as the number of multiplications used, or else we will be able to obtain such an algorithm by only slightly increasing the number of multiplications.

To find lower bounds, we rely on results described in [14]. We begin by defining notation. Computations are performed over the field H. The set of inputs to an algorithm is denoted by $B \subset H$. The elements of B are referred to as *indeterminates*. We also assume that we have at our disposal a ground field $G \subset H$. Typically, G is the smallest subfield of H, e.g., $H = \mathfrak{R}$ and $G = \mathcal{Q}$.

An *algorithm* is a sequence of elements from H, each element being either from B or the sum, difference, product, or quotient of two previous elements. An *m/d step* (informally referred to as simply a *multiplication*) is a step where the element from H does not come from B, and cannot be expressed as the linear combination over G of previous elements.

With this model, an algorithm can be thought of as a collection of field identities involving rational functions. Let $\mu\{f_1,\cdots,f_n\}$ be the minimum number of multiplications that any algorithm uses to compute $f_1,\cdots,f_n$ from B. To disallow data dependencies in $\mu\{f_1,\cdots,f_n\}$, the model does not allow branching, but it does allow division by a rational function that may take on zero values, so long as the function is not the zero function.

Let $L_G\{X; I\}$ denote the linear span over G of X in I, and let the quotient space of I over J be denoted by I/J. We then have the following theorem, which we state without proof.

THEOREM 2.1 [14]. $\mu\{f_1,\cdots,f_n\} \geqslant \dim(L_G\{f_1,\cdots,f_n; H/L_G\{B\}\})$.

Theorem 2.1 will be used in Sections 4–6. As we will always be computing matrix functions of matrices, we introduce the following simplified notation. Let $\mathbf{A}$ be a matrix, and let $\mathbf{F}(\mathbf{A})$ be a matrix function of $\mathbf{A}$. $L_G\{\mathbf{A}\}$ refers to the span of the set of elements of $\mathbf{A}$. Likewise, $\mu\{\mathbf{F}(\mathbf{A})\}$ is the number of scalar multiplications required to compute the elements of $\mathbf{F}$ from the elements of $\mathbf{A}$. Finally, $L_G\{\mathbf{F}(\mathbf{A})\}$ means $L_G\{\mathbf{F}(\mathbf{A}); H/L_G\{\mathbf{A}\}\}$.

3. RELATIONSHIP OF TOEPLITZ PRODUCTS AND INVERSES TO OUTER PRODUCTS

In this section, we demonstrate the relation of the computation of inverses and products of Toeplitz matrices to the computation of sums of certain structured outer products.

We begin with multiplication. Let $\mathbf{A}$ and $\mathbf{B}$ be $(n+1)\times(n+1)$ Toeplitz matrices. define the first row and column of $\mathbf{A}$ as $(a_0 \ \mathbf{a}_+^t)$ and $(a_0 \ \mathbf{a}_-^t)^t$, with

$$\mathbf{a}_+^t \equiv \begin{pmatrix} a_1 & a_2 & \cdots & a_n \end{pmatrix}$$

and

$$\mathbf{a}_-^t \equiv \begin{pmatrix} a_{-1} & a_{-2} & \cdots & a_{-n} \end{pmatrix}.$$

Also define the reverse vectors $\tilde{\mathbf{a}}_+$ and $\tilde{\mathbf{a}}_-$ as

$$\tilde{\mathbf{a}}_+^t \equiv \begin{pmatrix} a_n & a_{n-1} & \cdots & a_1 \end{pmatrix}$$

and

$$\tilde{\mathbf{a}}_-^t \equiv \begin{pmatrix} a_{-n} & a_{-n+1} & \cdots & a_{-1} \end{pmatrix}.$$

Use similar notation for the first column and row of $\mathbf{B}$.

Because $\mathbf{A}$ and $\mathbf{B}$ are Toeplitz matrices, they can be partitioned as

$$\mathbf{A} \equiv \begin{pmatrix} a_0 & \mathbf{a}_+^t \\ \mathbf{a}_- & \mathscr{A} \end{pmatrix} = \begin{pmatrix} \mathscr{A} & \tilde{\mathbf{a}}_+ \\ \tilde{\mathbf{a}}_-^t & a_0 \end{pmatrix}$$

and

$$\mathbf{B} \equiv \begin{pmatrix} b_0 & \mathbf{b}_+^t \\ \mathbf{b}_- & \mathscr{B} \end{pmatrix} = \begin{pmatrix} \mathscr{B} & \tilde{\mathbf{b}}_+ \\ \tilde{\mathbf{b}}_-^t & b_0 \end{pmatrix}.$$

Multiply the first partitions to obtain

$$\mathbf{AB} = \begin{pmatrix} a_0 b_0 + \mathbf{a}_+^t \mathbf{b}_- & a_0 \mathbf{b}_+^t + \mathbf{a}_+^t \mathscr{B} \\ \mathbf{a}_- b_0 + \mathscr{A} \mathbf{b}_- & \mathbf{a}_- \mathbf{b}_+^t + \mathscr{A}\mathscr{B} \end{pmatrix}.$$

Therefore,

$$(\mathbf{AB})_{i+1,j+1} = \left(\mathbf{a}_- \mathbf{b}_+^t + \mathscr{A}\mathscr{B}\right)_{i,j}, \qquad i,j = 0,\ldots,n-1. \tag{2}$$

Multiplying out the second partitions for $\mathbf{A}$ and $\mathbf{B}$ will similarly yield

$$(\mathbf{AB})_{i,j} = \left(\mathscr{A}\mathscr{B} + \tilde{\mathbf{a}}_+ \tilde{\mathbf{b}}_-^t\right)_{i,j}, \qquad i,j = 0,\ldots,n-1. \tag{3}$$

Combining Equations (2) and (3), we obt...

$$(\mathbf{AB})_{i+1,j+1} = (\mathbf{AB})_{i,j} + \left(\mathbf{a}_- \mathbf{b}_+^t - \tilde{\mathbf{a}}_+ \tilde{\mathbf{b}}_-^t\right)_{i,j}, \qquad i,j = 0,\ldots,n-1. \tag{4}$$

Equation (4) shows how to compute $\mathbf{AB}$ from its border via the computation of the sum of outer products $\mathbf{a}_- \mathbf{b}_+^t - \tilde{\mathbf{a}}_+ \tilde{\mathbf{b}}_-^t$. This equation can be generalized to the product of rectangular Toeplitz matrices. A consequence of Equation (4) is that $\mathbf{AB}$ is a Toeplitz matrix if and only if $\mathbf{a}_- \mathbf{b}_+^t - \tilde{\mathbf{a}}_+ \tilde{\mathbf{b}}_-^t = \mathbf{0}$. This fact has been observed in [13].

We now move on to inversion. The Trench algorithm [14, 3] is a two-stage algorithm for computing the inverse of a Toeplitz matrix. In the first stage, the border of the inverse is computed. The interior of the inverse is computed from its border in the second stage. Each stage requires $O(n^2)$ multiplications.

Several other algorithms exist for computing the border of the inverse. Among the more recently introduced are a class of "doubling" algorithms. Practical versions of these algorithms require $O(n \log^2 n)$ multiplications and a similar number of additions [1, 2, 17]. It is also possible to implement the algorithms using only $O(n \log n)$ multiplications but many more additions [17].

We are therefore concerned only with the second stage of the algorithm. Specifically, let $\mathbf{A}$ be given as in Equation (1), and let $\mathbf{B}$ be its inverse. The Trench algorithm allows for the computation of $\mathbf{B}$ via the relationship

$$\mathbf{B}_{i+1,j+1} = \mathbf{B}_{i,j} + \frac{1}{b_0}\left(\mathbf{b}_-\mathbf{b}_+^t - \tilde{\mathbf{b}}_+\tilde{\mathbf{b}}_-^t\right)_{i,j}, \qquad i,j = 0,\dots,n-1. \tag{5}$$

Gohberg and Semencul [6, 7] obtained an expression for $\mathbf{B}$ as the sum of products of triangular Toeplitz matrices:

$$b_0\mathbf{B} = \begin{pmatrix} b_0 & 0 & \cdots & 0 \\ b_1 & b_0 & \cdots & 0 \\ \vdots & \vdots & & \vdots \\ b_n & b_{n-1} & \cdots & b_0 \end{pmatrix}\begin{pmatrix} b_0 & b_{-1} & \cdots & b_{-n} \\ 0 & b_0 & \cdots & b_{1-n} \\ \vdots & \vdots & & \vdots \\ 0 & 0 & \cdots & b_0 \end{pmatrix}$$

$$-\begin{pmatrix} 0 & \cdots & 0 & 0 \\ b_{-n} & \cdots & 0 & 0 \\ \vdots & & \vdots & \vdots \\ b_{-1} & \cdots & b_{-n} & 0 \end{pmatrix}\begin{pmatrix} 0 & \cdots & b_2 & b_1 \\ 0 & \cdots & b_3 & b_2 \\ \vdots & & \vdots & \vdots \\ 0 & \cdots & 0 & 0 \end{pmatrix}. \tag{6}$$

which is algebraically equivalent to the Trench formula.

Equation (5) or (6) imposes a rigid structure on the form of the inverse of a Toeplitz matrix. In Section 6, we will find a lower bound for the number of multiplications required to invert a Toeplitz matrix. In so doing, the question arises whether the inverse of such a structured matrix is necessarily Toeplitz. This question is answered by the following proposition.

PROPOSITION 3.1. *Let $\mathbf{B}$ be an $(n+1)\times(n+1)$ matrix of the form given by Equation (5) or (6). Assume further that $b_0 \neq 0$ and that $\mathbf{A} \equiv \mathbf{B}^{-1}$ exists. Then $\mathbf{A}$ is a Toeplitz matrix.*

The proof of Proposition 3.1 is given in Appendix A.

Our proof uses the Trench formula and is quite simple. A related result, from which Proposition 3.1 can be easily derived, was shown in [7, Theorem 18.5], but, because it uses the Gohberg-Semencul formula, it is more involved then our derivation.

Note the similarity between Equations (4) and (5); in both, we obtain the desired matrix from its border and the sum of two outer products. We denote the border of a matrix $\mathbf{A}$ by $\partial\mathbf{A}$. From Equations (4) and (5), we see that we

TABLE 1

TOEPLITZ MATRIX PROBLEMS AND RELATED OUTER PRODUCTS[a]

$(n+1)\times(n+1)$ Toeplitz matrices	$n\times n$ outer product
AB	$\mathbf{ab}^t + \mathbf{cd}^t$
AB, symmetric	$\mathbf{ab}^t - \tilde{\mathbf{a}}\tilde{\mathbf{b}}^t$
$\mathbf{A}^2$	$\mathbf{ab}^t - \tilde{\mathbf{b}}\tilde{\mathbf{a}}^t$
$\mathbf{A}^2$, symmetric	$\mathbf{aa}^t - \tilde{\mathbf{a}}\tilde{\mathbf{a}}^t$
$\mathbf{A}^{-1}$	$\mathbf{ab}^t - \tilde{\mathbf{b}}\tilde{\mathbf{a}}^t$
$\mathbf{A}^{-1}$, symmetric	$\mathbf{aa}^t - \tilde{\mathbf{a}}\tilde{\mathbf{a}}^t$

[a]**A** and **B** are arbitrary (symmetric) Toeplitz matrices, and **a**, **b**, **c**, and **d** are arbitrary vectors.

may view $\partial\mathbf{A}$ as the first (or last) row and column of **A**. Alternatively, $\partial\mathbf{A}$ can be taken as the first and last row (or column) of **A**. Any of these four alternatives allows one to obtain either **AB** or $\mathbf{A}^{-1}$ from its border and two outer products.

If we impose additional structure on the matrices involved, we will obtain additional structure in the outer products. If the Toeplitz matrix that we wish to invert is symmetric,[1] then its inverse will be symmetric, so we will have $\mathbf{b}_+ = \mathbf{b}_-$ in Equation (5). Similar structure is added to matrix multiplication if the matrices are both symmetric and Toeplitz. If we wish to compute the square of a Toeplitz matrix, we will have $\mathbf{a}_+ = \mathbf{b}_+$ and $\mathbf{a}_- = \mathbf{B}_-$ in Equation (4). The exact structure of the sums of outer products for each problem is shown in Table 1.

4. COMPLEXITY OF THE OUTER PRODUCTS

In the previous section, we demonstrated the connection between Toeplitz matrix problems and four structured $n \times n$ outer products: $\mathbf{ab}^t + \mathbf{cd}^t$, $\mathbf{ab}^t - \tilde{\mathbf{b}}\tilde{\mathbf{a}}^t$, $\mathbf{ab}^t - \tilde{\mathbf{a}}\tilde{\mathbf{b}}^t$, and $\mathbf{aa}^t - \tilde{\mathbf{a}}\tilde{\mathbf{a}}^t$. The advantage of doing this is that the complexity analysis of the outer product problems is relatively simple.

In this section, we carry out the analysis of the multiplicative complexity of the outer products. It will be shown that all of the problems possess $O(n^2)$ multiplicative complexity, and that the more structured outer products require

[1]The structure added by symmetry in a Toeplitz matrix is equivalent to the structure added by persymmetry in a Hankel matrix.

fewer multiplications than the less structured ones. To simplify the discussion, we will henceforth assume that n is even. The analysis for n odd is very similar. When results differ, the results for n odd will be put in parentheses.

The elements of sums of outer products are quadratic forms; that is, their elements can be written as $\Sigma\alpha_{i,j}q_iq_j$, where $\alpha_{i,j}\in G$, and the q_i's are elements of the vectors **a**, **b**, **c**, and **d**. To obtain lower bounds on the number of multiplications required to compute these sums of outer products, we will use the results of Section 2. If the q_i's are indeterminates, then different quadratic terms are linearly independent of each other and the q_i's. We now find bounds for each of the four cases:

1. $\mathbf{ab}^t + \mathbf{cd}^t = \{a_{i+1}b_{j+1} + c_{i+1}d_{j+1}\}$. The (ij)th element is the only element with the term $a_{i+1}b_{j+1}$, so all elements are independent. To calculate $\mathbf{ab}^t + \mathbf{cd}^t$, we use the matrix multiplication algorithm from [16]. Compute a_ic_i and b_id_i for $i = 1,\dots,n$ ($2n$ μ's). We can now obtain output elements at one multiplication each through

$$a_ib_j + c_id_j = (a_i + d_j)(c_i + b_j) - a_ic_i - b_jd_j. \tag{7}$$

It follows that $n^2 \leqslant \mu\{\mathbf{ab}^t + \mathbf{cd}^t\} \leqslant n^2 + 2n$.

2. $\mathbf{ab}^t - \tilde{\mathbf{b}}\tilde{\mathbf{a}}^t = \{a_{i+1}b_{j+1} - b_{n-i}a_{n-j}\}$. The elements for which $i + j < n - 1$ are independent, as only the (ij)th element has the term $a_{i+1}b_{j+1}$. To compute this matrix, proceed as above with appropriate substitutions. Thus "c_id_i" is obtained free from "a_ib_i". We use Equation (7) only when $i + j < n - 1$, as $(\mathbf{ab}^t - \tilde{\mathbf{b}}\tilde{\mathbf{a}}^t)_{i,j} = -(\mathbf{ab}^t - \tilde{\mathbf{b}}\tilde{\mathbf{a}}^t)_{n-1-j,n-1-i}$. Therefore, $(n^2 - n)/2 \leqslant \mu\{\mathbf{ab}^t - \tilde{\mathbf{b}}\tilde{\mathbf{a}}^t\} \leqslant (n^2 + n)/2$.

3. $\mathbf{ab}^t - \tilde{\mathbf{a}}\tilde{\mathbf{b}}^t$. Pairs of elements of this matrix can be computed as two point circular convolutions. Specifically,

$$\begin{pmatrix} (\mathbf{ab}^t - \tilde{\mathbf{a}}\tilde{\mathbf{b}}^t)_{i,j} \\ (\mathbf{ab}^t - \tilde{\mathbf{a}}\tilde{\mathbf{b}}^t)_{i,n-i-j} \end{pmatrix} = \begin{pmatrix} a_{i+1} & -a_{n-i} \\ -a_{n-i} & a_{i+1} \end{pmatrix} \begin{pmatrix} b_{j+1} \\ b_{n-j} \end{pmatrix}.$$

Therefore, computing $\mathbf{ab} - \tilde{\mathbf{a}}\tilde{\mathbf{b}}^t$ is equivalent to computing the following circular convolutions:

$$(a_{i+1}\ -a_{n-i})*(b_{j+1}\ b_{n-j}), \qquad i = 0,\dots,n-1, \quad j = 0,\dots,\frac{n}{2} - 1. \tag{8}$$

If, in (8), we replace i by $n - 1 - i$, we will only switch the order and the signs of the outputs of the circular convolution. Hence the only convolutions that have to be computed in (8) are those for which $i, j \leqslant n/2 - 1$. This leaves $n^2/4$ two point circular convolutions. A two point circular convolution can be computed with two multiplications [3]. We can therefore compute $\mathbf{ab}^t - \tilde{\mathbf{a}}\tilde{\mathbf{b}}^t$ in $n^2/2$ multiplications. It is easy to see that the outputs of the circular convolutions described by (8) for which $i, j \leqslant n/2 - 1$ are independent. Therefore, $\mu\{\mathbf{ab}^t - \tilde{\mathbf{a}}\tilde{\mathbf{b}}^t\} = n^2/2$ [$(n^2 - 1)/2$ for n odd].

4. $\mathbf{aa}^t - \tilde{\mathbf{a}}\tilde{\mathbf{a}}^t$. View this matrix as a special case of the previous matrix, with $\mathbf{a} = \mathbf{b}$. The outputs of (8) will then not change if we switch i and j, so we only need to compute those convolutions for which $j \leqslant i < n/2$. Also, when $i = j$ we only need one multiplication, as

$$\begin{pmatrix} a_{i+1} & -a_{n-1} \end{pmatrix} * \begin{pmatrix} a_{i+1} & a_{n-i} \end{pmatrix} = \Big(\big(a_{i+1} - a_{n-i}\big)\big(a_{i+1} + a_{n-1}\big) \quad 0 \Big).$$

The elements of $\mathbf{aa}^t - \tilde{\mathbf{a}}\tilde{\mathbf{a}}^t$ for which $i \geqslant j$ and $i > n - j - 1$ are independent, so $\mu\{\mathbf{aa}^t - \tilde{\mathbf{a}}\tilde{\mathbf{a}}^t\} = n^2/4$ [$(n^2 - 1)/4$ for n odd].

As a final note, let us say that for the inversion problem the outer product of interest is $(1/b_0)(\mathbf{b}_+\mathbf{b}_-^t \tilde{\mathbf{b}}_- \tilde{\mathbf{b}}_+)$. The division by b_0 requires no more than n additional multiplications, as we can first divide the elements of $\mathbf{b}_+$ by b_0 and then proceed as above. Because b_0 is not among the elements of $\mathbf{b}_+$ or $\mathbf{b}_-$, the dimension of the span of the outer product is not affected by the division by b_0.

5. TOEPLITZ MATRIX MULTIPLICATION

Equation (4) relates the computation of $\mathbf{AB}$ to the computation of $\partial(\mathbf{AB})$ and $\mathbf{a}_-\mathbf{b}_+^t - \tilde{\mathbf{a}}_+\tilde{\mathbf{b}}_-^t$. The relatively simple structure of $\partial(\mathbf{AB})$ and $\mathbf{a}_-\mathbf{b}_+^t - \tilde{\mathbf{a}}_+\tilde{\mathbf{b}}_-^t$ will be exploited in this section to obtain lower and upper bounds on $\mu\{\mathbf{AB}\}$. From Equation (4) we can readily see that

$$\mu\{\mathbf{AB}\} \geqslant \dim\Big(L_G\big\{\{\mathbf{a}_-\mathbf{b}_+^t - \tilde{\mathbf{a}}_+\tilde{\mathbf{b}}_-^t\} \cup \partial(\mathbf{AB})\big\}\Big) \tag{9}$$

and

$$\mu\{\mathbf{AB}\} \leqslant \mu\{\mathbf{a}_-\mathbf{b}_+^t - \tilde{\mathbf{a}}_+\tilde{\mathbf{b}}_-^t\} + \mu\{\partial(\mathbf{AB})\}. \tag{10}$$

We will consider lower bounds first. For definiteness, we will take $\partial(\mathbf{AB})$ as the first and last row of $\mathbf{AB}$; that is, as $(a_0 \ \mathbf{a}_+^t)\mathbf{B}$ and $(\tilde{\mathbf{a}}_-^t \ a_0)\mathbf{B}$. We now evaluate the right hand side of Equation (9).

The term $a_0 b_i$ has unit weight in the ith element of $(a_0\ \mathbf{a}_+^t)\mathbf{B}$, while it has zero weight in the other elements of $(a_0\ \mathbf{a}_+^t)\mathbf{B}$, in all the elements of $\mathbf{a}_-\mathbf{b}_+^t - \tilde{\mathbf{a}}_+\tilde{\mathbf{b}}_-^t$, and in the first n elements of $(\tilde{\mathbf{a}}_-^t\ a_0)\mathbf{B}$. Likewise, the term $a_0 b_{i-n}$ has unit weight in the ith element of $(\tilde{\mathbf{a}}_-^t\ a_0)\mathbf{B}$, while it has, for $i < n$, zero weight in the other elements of $(\tilde{\mathbf{a}}_-^t\ a_0)\mathbf{B}$, in all the elements of $\mathbf{a}_-\mathbf{b}_+^t - \tilde{\mathbf{a}}_+\tilde{\mathbf{b}}_-^t$, and in all the elements of $(a_0\ \mathbf{a}_+^t)\mathbf{B}$. Therefore, the first n elements of $(\tilde{\mathbf{a}}_-^t\ a_0)\mathbf{B}$ and all the elements of $\mathbf{a}_-\mathbf{b}_+^t - \tilde{\mathbf{a}}_+\tilde{\mathbf{b}}_-^t$ and $(a_0\ \mathbf{a}_+^t)\mathbf{B}$ each have at least one different quadratic element. From Equation (9) and Theorem 2.1, we can conclude that $\mu\{\mathbf{AB}\} \geqslant n^2 + 2n + 1$.

In Section 4, we found an algorithm to compute $\mathbf{a}_-\mathbf{b}_+^t - \tilde{\mathbf{a}}_+\tilde{\mathbf{b}}_-^t$ (note that, as the vectors of $\mathbf{a}_-\mathbf{b}_+^t - \tilde{\mathbf{a}}_+\tilde{\mathbf{b}}_+^t$ are arbitrarily related, computing $\mathbf{a}_-\mathbf{b}_+^t - \tilde{\mathbf{a}}_+\tilde{\mathbf{b}}_-^t$ is equivalent to computing $\mathbf{ab}^t + \mathbf{cd}^t$). In order to find an upper bound on $\mu\{\mathbf{AB}\}$, we need only to find an algorithm to compute the product of vectors and a Toeplitz matrix. It is shown in [8] that this can be done in $2n + 1$ multiplications. The algorithm presented in [8] uses an excessive number of additions when n is large. To avoid the large number of additions, we can extend $\mathbf{B}$ to a circulant matrix of size $2n + 1$ or larger. Then $\partial(\mathbf{AB})$ can be computed as two $2n + 1$ point (or larger) circular convolutions. These can be computed with fast Fourier transforms (FFTs) in $O(n \log n)$ additions and $O(n \log n)$ multiplications [3]. Therefore, the computation of $\partial(\mathbf{AB})$ requires no more than $4n + 2$ multiplications, and a practical algorithm to compute $\partial(\mathbf{AB})$ can be found that uses $O(n \log n)$ multiplications. Equation (10) now allows us to find an upper bound on $\mu\{\mathbf{AB}\}$ (see Table 2).

We can use the above method to find lower and upper bounds on $\mu\{\mathbf{AB}\}$ in the special cases $\mathbf{A} = \mathbf{B}$ and/or $\mathbf{A}$ and $\mathbf{B}$ symmetric. First, we find the number of elements in the border that are linearly independent of each other and the linearly independent elements of the related sums of outer products

TABLE 2

MULTIPLICATIVE COMPLEXITY OF TOEPLITZ PRODUCTS[a]

		$\mu\{\cdot\}$		
Problem	n	Lower bound	Upper bound	Practical algorithm
$\mathbf{AB}$		$n^2 + 2n + 1$	$n^2 + 6n + 2$	$n^2 + O(n \log n)$
$\mathbf{AB}$, sym.	Even	$(n^2 + 2n + 2)/2$	$(n^2 + 4n + 2)/2$	$n^2/2 + O(n \log n)$
$\mathbf{AB}$, sym.	Odd	$(n^2 + 2n + 1)/2$	$(n^2 + 4n - 1)/2$	$n^2/2 + O(n \log n)$
$\mathbf{A}^2$		$(n^2 + 3n + 2)/2$	$(n^2 + 9n + 4)/2$	$n^2/2 + O(n \log n)$
$\mathbf{A}^2$, sym.	Even	$(n^2 + 4n)/4$	$(n^2 + 8n + 4)/4$	$n^2/4 + O(n \log n)$
$\mathbf{A}^2$, sym.	Odd	$(n^2 + 4n - 1)/4$	$(n^2 + 8n - 1)/4$	$n^2/4 + O(n \log n)$

[a]A and B are arbitrary $(n + 1) \times (n + 1)$ (symmetric) Toeplitz matrices.

(see Table 1). The number of such elements plus the number of linearly independent terms in the sum of outer products is a lower bound on the number of multiplication required.

In finding upper bounds, we try to take advantage of any added structure to reduce the number of multiplications used to compute $\partial(\mathbf{AB})$. Specifically, when $\mathbf{A}$ and $\mathbf{B}$ are symmetric, we have $\mathbf{a}_- = \mathbf{a}_+$ and $\mathbf{B} = \mathbf{B}^t$, so that $(\tilde{\mathbf{a}}^t_-\ a_0)\mathbf{B} = [(a_0\ \mathbf{a}^t_+)\mathbf{J}](\mathbf{JBJ}) = [(a_0\ \mathbf{a}^t_+)\mathbf{B}]\mathbf{J}$. The matrix $\mathbf{J}$ is the anti-diagonal matrix. Hence $\partial(\mathbf{AB})$ is specified by the first column of $\mathbf{AB}$. Also, when $\mathbf{B}$ is symmetric and n is odd, we can save one multiplication when computing $(a_0\ \mathbf{a}^t_+)\mathbf{B}$ [14]. As Table 1 makes clear, the condition $\mathbf{A} = \mathbf{B}$ and/or $\mathbf{A}$ and $\mathbf{B}$ symmetric also adds structure to the related outer product. We know from Section 4 that this structure allows us to find faster algorithms to compute the related outer products. The resulting upper and lower bounds for $\mu\{\mathbf{AB}\}$ are presented in Table 2.

6. TOEPLITZ MATRIX INVERSION

We will find upper bounds for the number of multiplications needed to compute $\mathbf{A}^{-1}$ in essentially the same way that we found upper bounds on the multiplicative complexity of $\mathbf{AB}$. Upper bounds will come from adding the number of multiplications used to compute the border of $\mathbf{A}^{-1}$ to the number used to compute the related outer product. Lower bounds will come from estimating the dimension of the span of the output.

As noted in Section 3, $\partial\mathbf{A}^{-1}$ can be computed with $O(n \log n)$ multiplications and many more additions, or $O(n \log^2 n)$ multiplications and a similar number of additions. Because $\mathbf{b}_-\mathbf{b}^t_+ - \tilde{\mathbf{b}}_+\tilde{\mathbf{b}}^t_-$ can be computed with $(n^2 + n)/2$ multiplications, computing all of $\mathbf{A}^{-1}$ requires at most $n^2/2 + O(n \log n)$ multiplications. (Note: The algorithms presented in [1, 2, 17] are only valid when the leading submatrices of $\mathbf{A}$ are nonsingular. Also, the Trench formula [Equation (5)] is only valid when $b_0 \neq 0$. However, these singular cases do not change the formal arithmetic complexity of the problems.)

To find a lower bound on the number of multiplications required to compute $\mathbf{A}^{-1}$, we wish to find $\dim(L_G\{\mathbf{A}^{-1}; H/\mathbf{A}\})$. From Proposition 3.1, we see that the mapping of the border of $\mathbf{A}$ to the border of its inverse is bijective. Thus, the $2n + 1$ border elements of $\mathbf{B}$ can be viewed as virtual indeterminates. From case 2 in Section 4, we see that

$$\dim\left(L_G\left\{\partial\mathbf{B} \cup \left(\mathbf{b}_-\mathbf{b}^t_+ - \tilde{\mathbf{b}}_+\tilde{\mathbf{b}}^t_-\right); H\right\}\right) = \frac{n^2 - n}{2} + 2n + 1.$$

TABLE 3
MULTIPLICATIVE COMPLEXITY OF INVERTING AN $(n+1)\times(n+1)$ TOEPLITZ MATRIX

	$\mu\{\cdot\}$		
Symmetric	Lower bound	Upper bound	Practical algorithm
No	$n(n-1)/2$	$n^2/2+O(n\log n)$	$n^2/2+O(n\log^2 n)$
Yes	$n^2/4[(n^2-1)/4,\ n \text{ odd}]$	$n^2/4+O(n\log n)$	$n^2/4+O(n\log^2 n)$

Because there are only $2n+1$ distinct elements in A, it follows that

$$\dim\left(L_G\left\{\partial\mathbf{B}\cup\left(\mathbf{b}_-\mathbf{b}_+^t-\tilde{\mathbf{b}}_+\tilde{\mathbf{b}}_-^t\right);H/L_G\{\mathbf{A}\}\right\}\right)=\frac{n^2-n}{2}.$$

We can now use Equation (5) and Theorem 2.1 to deduce that

$$\mu\{\mathbf{A}^{-1}\}\geqslant\dim\left(L_G\{\mathbf{A}^-;H/L_G\{\mathbf{A}\}\}\right)\geqslant\frac{n(n-1)}{2}. \tag{11}$$

The nature of this lower bound is slightly different from that of the lower bounds of the last two sections. In the previous cases, we found the exact dimension of the span of the outputs, which we used as a lower bound on the multiplicative complexity. Here, we have only found a lower bound on the dimension of the span of the outputs, which in turn bounds $\mu\{\mathbf{A}^{-1}\}$. However, finding an exact value for $\dim(L_G\{\mathbf{A}^{-1};H/L_G\{\mathbf{A}\}\})$ cannot improve the bound given by Equation (11) by more than $2n+1$ multiplications.

The results of this section can be extended to A symmetric in a straightforward manner. The border of A (or $\mathbf{A}^{-1}$) is now taken to be its first column. The related outer product is $\mathbf{b}_+\mathbf{b}_+^t-\tilde{\mathbf{b}}_+\tilde{\mathbf{b}}_+^t$ (case 4 of Section 4). We can now proceed as above. Upper and lower bounds for computing $\mathbf{A}^{-1}$ for the symmetric and general cases are listed in Table 3.

7. CONCLUSION

This paper has shown that a relationship similar to Trench's formula exists between the border and the interior of the product of Toeplitz matrices. By using the Gohberg-Semencul formula, we can express the inverse of a Toeplitz matrix as the sum of two lower/upper Toeplitz products. An important difference between Toeplitz products and inverses is that no such formula exists for Toeplitz products, as the so-called *displacement rank* of a Toeplitz product is four [5].

Both Toeplitz products and inverses were seen to be obtainable from their borders through the sum of two outer products, and the added structure for cases when the matrices have additional imposed structure was also noted. Practical algorithms were given for all problems.

By deriving lower bounds for the problems, it was shown that any algorithm used to calculate the product or inverse of Toeplitz matrices must use $O(n^2)$ multiplications. In fact, all the algorithms that were given are asymptotically optimal, in the sense that the ratio of multiplications used to the derived lower bounds tends to unity as n tends to infinity.

The fact that these problems are lower bounded by $O(n^2)$ is indeed quite surprising when compared to general matrices. For general matrices, the order of multiplicative complexity is known to be the same for matrix inversion, system solution, calculation of determinants, and matrix multiplication [12]. In contrast, the order of complexity for inverting a Toeplitz matrix is higher than for solving a Toeplitz system of equations. Moreover, while calculating the inverse of a Toeplitz matrix requires $O(n^2)$ multiplications, the algorithm given in [1] can be used to calculate the determinant of a Toeplitz matrix in $O(n \log n)$ multiplications and many more additions, or $O(n \log^2 n)$ additions and multiplications.

It is also interesting to compare $\mu(\mathbf{AM})$ with $\mu(\mathbf{AB})$, where **A** and **B** are Toeplitz matrices and **M** is a general matrix. The product of a Toeplitz matrix and a general matrix can be computed with $O(n^2)$ multiplications (using n Toeplitz-vector products), so the order of the multiplicative complexity is not reduced when the second matrix is Toeplitz. However, if we use n optimal Toeplitz-vector products, we will use more than $O(n^2)$ additions.

In this paper, we have dealt with general Toeplitz matrices. The situation simplifies considerably when we deal with lower (or upper) triangular Toeplitz matrices. Lower triangular Toeplitz matrices are closed under multiplication and inversion. Obtaining the product of two such matrices then reduces to computing the product of a Toeplitz matrix and a vector. Also, the inverse of a lower triangular Toeplitz matrix can be computed in $O(n)$ multiplications [8]. The techniques developed in this paper can be used to show that the multiplicative complexity of multiplying an upper triangular Toeplitz matrix by a lower triangular Toeplitz matrix is $n^2 + O(n)$. [A practical algorithm would use $n^2 + O(n \log n)$.]

An important open question is finding the order of the multiplicative complexity of computing the border of the inverse of a general (or symmetric) Toeplitz matrix. Because the border has only $2n + 1$ elements, direct use of Theorem 2.1 cannot give a lower bound that is any better than $O(n)$. (In fact, by considering the special case when **A** is circulant, we can use Theorem 2.1

to obtain a lower bound of $n + 1$ multiplications.) Therefore, either a different lower bounding technique will have to be used or a more efficient algorithm devised before the multiplicative complexity of this problem can be determined.

APPENDIX A. PROOF OF PROPOSITION 3.1

We begin by showing that $\mathbf{B}$ is persymmetric, i.e., that $\mathbf{B}_{i,j} = \mathbf{B}_{n-j,n-i}$. Without loss of generality, we can assume $i + j < n$. Then from Equation (5), we have

$$\begin{aligned}
\mathbf{B}_{i,j} - \mathbf{B}_{n-j,n-i} &= \sum_{k=0}^{n-j-i-1} \left(\mathbf{b}_-\mathbf{b}_+^t - \tilde{\mathbf{b}}_+\tilde{\mathbf{b}}_-^t\right)_{i+k,j+k} \\
&= \sum_{k=0}^{n-j-i-1} \left(b_{-i-k-1}b_{j+k+1} - b_{n-i-k}b_{-n+j-k}\right) \\
&= \sum_{k=0}^{n-j-i-1} b_{-i-k-1}b_{j+k+1} - \sum_{l=n-j-i-1}^{0} b_{j+l+1}b_{-i-l-1} \\
&= 0.
\end{aligned}$$

Define $\mathbf{C} \in \Re^{n\times n}$ as

$$\mathbf{C}_{i,j} \equiv \mathbf{B}_{i,j} - \frac{1}{b_0}\left(\tilde{\mathbf{b}}_+\tilde{\mathbf{b}}_-^t\right)_{i,j}, \qquad i,j = 0,\ldots,n-1.$$

Then

$$\mathbf{B}_{i,j} = \mathbf{C}_{i,j} + \frac{1}{b_0}\left(\tilde{\mathbf{b}}_+\tilde{\mathbf{b}}_-^t\right)_{i,j}, \qquad i,j = 0,\ldots,n-1. \tag{12}$$

Equations (5) and (12) imply

$$\mathbf{B}_{i+1,j+1} = \mathbf{C}_{i,j} + \frac{1}{b_0}\left(\mathbf{b}_-\mathbf{b}_+^t\right)_{i,j} \qquad i,j = 0,\ldots,n-1. \tag{13}$$

We now use Equations (12) and (13) and the fact that $\mathbf{B}$ is persymmetric to partition $\mathbf{B}$ in two ways:

$$\mathbf{B} = \begin{pmatrix} b_0 & \mathbf{b}_+^t \\ \mathbf{b}_- & \mathbf{C} + \frac{1}{b_0}\mathbf{b}_-\mathbf{b}_+^t \end{pmatrix} = \begin{pmatrix} \mathbf{C} + \frac{1}{b_0}\tilde{\mathbf{b}}_+\tilde{\mathbf{b}}_-^t & \tilde{\mathbf{b}}_+ \\ \tilde{\mathbf{b}}_-^t & b_0 \end{pmatrix}. \tag{14}$$

Partition $\mathbf{A}$ in a similar manner,

$$\mathbf{A} \equiv \begin{pmatrix} a_0 & \mathbf{a}_1^t \\ \mathbf{a}_2 & \mathscr{A} \end{pmatrix} = \begin{pmatrix} \mathscr{A}' & \mathbf{a}_3^t \\ \mathbf{a}_4 & a_0 \end{pmatrix}. \tag{15}$$

By definition, $\mathbf{BA} = \mathbf{I}_{n(n+1)\times(n+1)}$. Block multiplying the first partitions for $\mathbf{B}$ and $\mathbf{A}$, we obtain

$$\begin{pmatrix} a_0 b_0 + \mathbf{b}_+^t\mathbf{a}_2 & b_0\mathbf{a}_1^t + \mathbf{b}_+^t\mathscr{A} \\ \mathbf{b}_- a_0 + \frac{1}{b_0}\mathbf{b}_-\mathbf{b}_+^t\mathbf{a}_2 + \mathbf{C}\mathbf{a}_2 & \mathbf{b}_-\mathbf{a}_1^t + \mathbf{C}\mathscr{A} + \frac{1}{b_0}\mathbf{b}_-\mathbf{b}_+^t\mathscr{A} \end{pmatrix}$$

$$= \begin{pmatrix} 1 & \mathbf{0}_{1\times n} \\ \mathbf{0}_{n\times 1} & \mathbf{I}_{n\times n} \end{pmatrix}.$$

Therefore,

$$\mathbf{b}_-\mathbf{a}_1^t + \mathbf{C}\mathscr{A} + \frac{1}{b_0}\mathbf{b}_-\mathbf{b}_+^t\mathscr{A} = \mathbf{I}_{n\times n}$$

and

$$b_0\mathbf{a}_1^t + \mathbf{b}_+^t\mathscr{A} = \mathbf{0}_{1\times n}.$$

Hence,

$$\mathbf{C}\mathscr{A} = \mathbf{I}_{n\times n},$$

so that $\mathscr{A} = \mathbf{C}^{-1}$.

In an analogous manner, we can block multiply the second partitions of $\mathbf{B}$ and $\mathbf{A}$ given by Equations (14) and (15). This will result in the conclusion that

$\mathscr{A}' = \mathbf{C}^{-1}$. It follows that $\mathscr{A} = \mathscr{A}'$. From Equation (15), we can now see that $\mathbf{A}$ is a Toeplitz matrix, proving Proposition 3.1.

The authors wish to thank Dr. Ephraim Feig of IBM Thomas J. Watson Research center for comments on earlier versions of the manuscript, among other things bringing [11] to our attention and mentioning the algorithm for the first case in Section 4.

REFERENCES

1 G. S. Ammar and W. B. Gragg, Superfast solution of real positive definite Toeplitz systems, *SIAM J. Matrix Anal. Appl.* 9:61–76 (1988).
2 R. R. Bitmead and B. D. O. Anderson, Asymptotically fast solution of Toeplitz and related systems of linear equations, *Linear Algebra Appl.* 34:103–116 (1980).
3 R. E. Blahut, *Fast Algorithms for Digital Signal Processing*, Addison-Wesley, Reading, Mass., 1984.
4 J. R. Bunch, Stability of methods for solving Toeplitz systems of equations, *SIAM J. Sci. Statist. Comput.* 6:349–364 (1985).
5 B. Friedlander et al., New inversion formulas for matrices classified in terms of their distance from Toeplitz matrices, *Linear Algebra Appl.* 27:31–60 (1979).
6 I. C. Gohberg and I. A. Fel'dman, *Convolution Equations and Projection Methods for Their Solution*, Amer. Math. Soc., Providence, 1974.
7 I. S. Iohvidov, *Hankel and Toeplitz Matrices and Forms*, Birkhäuser, Boston, 1982.
8 J. C. Lafon, Base tensorielle des matrices de Hankel (ou de Toeplitz) applications, *Numer. Math.* 23:349–361 (1974).
9 N. Levinson, The Wiener rms error criterion in filter design and prediction, *J. Math. Phys.* 25:261–278 (1947).
10 E. Linzer, On the Stable Inversion of Banded Toeplitz Systems, Technical Report, the Center for Telecommunications Research, Columbia Univ., New York, to appear.
11 D. Makarov, The lower bound for the number of multiplications for algorithms for calculating the product of Hankel matrices, *U.S.S.R. Comput. Math. and Math. Phys.* 17:195–199 (1978).
12 N. Pippenger, Algebraic complexity theory, *IBM J. Res. Develop.* 25:825–832 (1981).
13 T. Shalom, On algebras of Toepliz matrices, *Linear Algebra Appl.* 96:211–226 (1987).
14 W. F. Trench, An algorithm for the inversion of finite Toeplitz matrices, *J. SIAM*, 12:512–522 (1964).

15 S. Winograd, *Arithmetic Complexity of Computations*, SIAM CBMS-NSF Ser., No. 33, SIAM, Philadelphia, 1980.

16 S. Winograd, A new algorithm for inner products, *IEEE Trans. Comput.* 17:693–694 (1968).

17 H. M. Zhang and P. Duhamel, Doubling Levinson/Schur algorithm and its implementation, in *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing*, 1989, pp. 1115–1118.

18 H. M. Zhang, Algorithms Rapides et Matrices du Toeplitz, Ph.D. Thesis, ENST, Paris, 1989.

Received 2 October 1989; final manuscript accepted 10 May 1990