MATHEMATICS OF COMPUTATION, VOLUME 28, NUMBER 126, APRIL 1974, PAGES 349–368

On Fourier-Toeplitz Methods for Separable Elliptic Problems

By D. Fischer, G. Golub,* O. Hald, C. Leiva,** and O. Widlund**

Abstract. Some very fast numerical methods have been developed in recent years for the solution of elliptic differential equations which allow for separation of variables. In this paper, a Fourier-Toeplitz method is developed as an alternative to the well-known methods of Hockney and Buneman. It is based on the fast Fourier transform and Toeplitz factorizations. The use of Toeplitz factorizations combined with the Sherman-Morrison formula is also systematically explored for linear systems of algebraic equations with band matrices of Toeplitz, or almost Toeplitz form. Finally, results of numerical experiments are described.

1. Introduction. In recent years, some very fast and accurate methods have been developed for the direct solution of the sparse systems of linear algebraic equations which arise when elliptic problems are solved by finite difference or finite element methods. Several of these algorithms implement, in very efficient ways, the idea of separation of variables. The best known of these are due to Hockney [15], [16] and Buneman [5],[6].

In this paper, we will present an alternative to Hockney's and Buneman's methods for the solution of elliptic problems with constant coefficients on rectangular regions and on infinite parallel strips. Our method, like Hockney's, is based on the use of the fast Fourier transform for one of the variables, but it uses an alternative way of solving the resulting systems of linear algebraic equations. These systems can be represented with band matrices of Toeplitz form or as low-rank perturbations of such matrices. The systems are solved by a combination of Toeplitz factorizations and applications of the Sherman-Morrison formula or its block version, the Woodbury formula (cf. Householder [17]). By using the Toeplitz structure, we are able to take greater advantage of the special structure of the matrices than when Gaussian elimination is used. This leads to a considerable saving in storage. We note that the odd-even reduction method has similar advantages but that it can be used only for a subset of the problems which we can handle. We include a discussion of the use of Toeplitz factorizations for more general band matrices. The method requires considerably much less storage than the Cholesky and Gauss elimination methods and it is also found to be quite competitive, in terms of arithmetic operations, in certain cases. For earlier work on Toeplitz methods, cf. Bakes [1], Bareiss [2], Evans [10], Evans

---

Received July 5, 1973.

AMS (MOS) subject classifications (1970). Primary 65F05, 65N20, 65T05; Secondary 15–04, 31C20, 47B35.

* The work of this author was supported in part by the U. S. Atomic Energy Commission, Contract No. AT(04-3)-326, Agreement No. 30, and the National Science Foundation, Grant GJ 35135X.

** The work of this author was supported in part by the U. S. Atomic Energy Commission, Contract AT(11–1)-3077 at the Courant Institute of Mathematical Sciences, New York University.

and Forrington [11], Malcolm and Palmer [18], and, for a somewhat different class of problems, Rose [22].

Separation of variables can be used only for regions which, after a possible change of independent variables, are rectangular and for differential operators of a special form. Similar restrictions are imposed on the discrete problems. For a discussion of the special structure which is needed for the use of these methods, cf. Widlund [25]. However, even for problems on nonrectangular regions, or with boundary conditions which do not allow for separation of variables, the fast methods can be used provided that the operators allow for separation of the variables on some appropriate region. The idea is to imbed the given region in a rectangle and combine a fast method with the Woodbury formula or a minimization algorithm. Cf. Buzbee, Dorr, George and Golub [8], George [14], Proskurowski and Widlund [19], and Widlund [25]. Proper implementations of the imbedding methods lead to a somewhat more favorable operation count than George's powerful ordering algorithm for Cholesky factorization [13]. That method is however applicable to a much wider class of positive definite, symmetric matrices.

The Fourier-Toeplitz method has been tried in a series of numerical experiments at the AEC Computing Center at the Courant Institute of Mathematical Sciences and at the Institute of Technology in Stockholm. Some of the results, which are reported in the last section, show that the method produces highly accurate solutions in a time which for the CDC 6600 is 60–80% of that of a program which implements Buneman's method.

The authors wish to thank Dr. R. Singleton of Stanford Research Institute for making his fast Fourier transform programs available, and Dr. B. Buzbee of the Los Alamos Laboratory for the use of his Buneman program.

2. Tri-Diagonal Problems. Our interest in one-dimensional problems results from our use of the separation of variables technique which reduces multi-dimensional to one-dimensional problems. The linear systems of equations under study also appear in other applications such as spline interpolation. We will discuss the solution of linear systems of algebraic equations with real band matrices and begin by considering the special case of the $n \times n$ tri-diagonal matrix

$$A = \begin{pmatrix} \lambda & -1 & & & & & \\ -1 & \lambda & -1 & & & & \\ & -1 & \ddots & \ddots & & & \\ & & \ddots & \ddots & \ddots & & \\ & & & \ddots & \ddots & \ddots & \\ & & & & \ddots & \lambda & -1 \\ & & & & & -1 & \lambda \end{pmatrix}, \quad \lambda \geq 2 .$$

The corresponding linear system could be solved by band Gauss elimination type methods (cf. Forsythe and Moler [12]), or by odd-even reduction (cf. Hockney [15] or Widlund [25]). The latter of these methods takes into account not only the band structure but also the fact that the matrix A has Toeplitz form, i.e., the values of its

elements a_{ij} depend only on $i - j$. Such matrices require very little storage and are easy to handle even in other respects. It is therefore natural to try to find a LU, or LDL^T, decomposition of A in terms of Toeplitz matrices. This is not possible for finite values of n. However, for a perturbed matrix B, we find

$$B = \begin{pmatrix} \mu & -1 & & & \\ -1 & \lambda & -1 & & \\ & -1 & \lambda & \ddots & \\ & & \ddots & \ddots & -1 \\ & & & -1 & \lambda \end{pmatrix}$$

$$= \begin{pmatrix} 1 & & & & \\ -\mu^{-1} & 1 & & & \\ & -\mu^{-1} & \ddots & & \\ & & \ddots & 1 & \\ & & & -\mu^{-1} & 1 \end{pmatrix} \begin{pmatrix} \mu & -1 & & & \\ & \mu & -1 & & \\ & & \ddots & \ddots & \\ & & & \ddots & -1 \\ & & & & \mu \end{pmatrix}$$

where $\mu = \lambda/2 \pm (\lambda^2/4 - 1)^{1/2}$. It is easy to see that the plus sign should be chosen, because we then have $\mu \geqq 1$ and thus diagonally dominant matrices. The numerical stability of the process for solving $LUx = f$, $B = LU$ follows immediately. If, on the other hand, the minus sign were chosen and $\lambda > 2$, then $0 < \mu < 1$ and we have to expect an exponential growth of round-off errors. This becomes apparent when we consider the two-term recursion relationships represented by the bi-diagonal matrices.

The change in the upper left-most element of A is compensated for by the use of the Sherman-Morrison formula. That is, if $A = B + uv^T$, with u and v column vectors and v^T denotes the transpose of v, then

$$A^{-1} = B^{-1} - B^{-1}u(1 + v^T B^{-1}u)^{-1}v^T B^{-1}.$$

The matrix uv^T is of rank one and, in this case, we can choose $u = (1, 0, \cdots, 0)^T$ and $v = (\lambda - \mu)u = (1/\mu)u$.

We remark that the Sherman-Morrison formula, and its block version, the Woodbury formula,

$$A^{-1} = B^{-1} - B^{-1}U(I_p + V^T B^{-1}U)^{-1}V^T B^{-1},$$

sometimes provide a useful tool to decide whether or not a matrix A is singular. Here, $A = B + UV^T$, U and V are $n \times p$ matrices and I_p the $p \times p$ identity matrix. Given that B is nonsingular, A is singular if and only if $I_p + V^T B^{-1}U$ is singular. Assume that $(I_p + V^T B^{-1}U)\varphi = 0$ for some nonzero vector φ. Then, since $V^T B^{-1}U\varphi = -\varphi$, the vector $B^{-1}U\varphi$ is different from zero. This vector is an eigenvector to A corresponding to the eigenvalue zero because

$$AB^{-1}U\varphi = (B + UV^T)B^{-1}U\varphi = U(I_p + V^T B^{-1}U)\varphi = 0.$$

Conversely, if $I_p + V^T B^{-1} U$ is nonsingular, the Woodbury formula provides an explicit formula for A^{-1}.

We also remark that the matrix A, studied above, might correspond to a standard second order accurate finite difference approximation to $-\partial_x^2 u + cu = f$, c some nonnegative constant, with Dirichlet boundary conditions. By an appropriate change of one of the boundary conditions, we arrive at a matrix B of the form above, with $\lambda = 2 + h^2 c$. To solve $Ax = b$, by our method, we find $B^{-1}b$ and add to it a solution of the special form const $\times B^{-1}u$. The second term is a correction term which makes the solution satisfy the correct boundary conditions. Alternatively, we can modify the data at one endpoint and use the boundary condition corresponding to the matrix B.

This method can be implemented in several ways. Here we will suggest a procedure which requires very little temporary storage. We will restrict our discussion to the case when $\lambda > 2$, i.e., $\mu > 1$. We start by computing the constant

$$d = (\lambda - \mu)(1 + v^T B^{-1} u)^{-1} \cdot (1 - \mu^{-2})^{-1}.$$

It is easy to see that $d = \mu^{-1}(1 - \mu^{-(2n+2)})^{-1}$. This part of the computation, which is independent of the particular data vector, can be carried out in a time comparable to a fixed number of arithmetic operations, provided we use an economical algorithm for the evaluation of the exponential function, taking advantage of the finite word length of the computer. If we do not wish to preserve the data, we can now execute the entire procedure in place, using only a fixed number of temporary storage locations. We first compute $B^{-1}b$ in the usual way and let it occupy the storage originally containing b. The elements of $B^{-1}b$ are thereafter modified one by one by successively subtracting the elements of two vectors, the sum of which equals the second term

$$y = B^{-1}u(1 + v^T B^{-1} u)^{-1} v^T B^{-1} b$$

of the Sherman-Morrison formula. It is elementary to verify that

$$y_\nu = (\mu^{-\nu} - \mu^{\nu-2n-2}) d (B^{-1}b)_1.$$

The first component should be computed recursively for increasing values of ν, to assure numerical stability, while the second component should be found for decreasing values of ν. The required number of operations are $4 + \mathcal{O}(1/n)$ multiplications/divisions and $4 + \mathcal{O}(1/n)$ additions/subtractions per unknown. If $B^{-1}u$ is computed and stored, we can save one fourth of this work if the same system of equations is solved many times with different data vectors.

We note (cf. Widlund [25]) that the operation counts for Gaussian elimination and odd-even reduction methods are somewhat more favorable in this special case. The correct choice of method will in fact depend on which computer, compilers, storage, etc., are available. Also compare the discussion at the end of Section 3.

Essentially the same method can be applied to other matrices which differ from B by only a few elements. It is sometimes advantageous to modify the algorithm if we want to change the elements in the lower right-hand corner of B. If, for example, we consider a matrix C which differs from B only in the element in the lower right-

hand corner, we can avoid using the Sherman-Morrison formula and instead use the regular LU factorization of C. We should then, of course, take advantage of the fact that the $(n-1)$st rows of L and $(n-1)$st columns of U are known from a factorization into Toeplitz factors of a submatrix of order $n-1$. It should also be clear from this discussion that, in certain cases, a UL Toeplitz factorization is preferable in order to minimize the rank of the modification matrix which is to be handled by the Woodbury formula. The implementation of the Woodbury formula is further discussed in Section 3. For yet another variant of the algorithm, compare our discussion of twisted Toeplitz factorizations in Section 4.

3. Toeplitz Factorization of General One-Dimensional Problems. We will now turn to a discussion of a general matrix which differs by no, or only a few, elements from a symmetric band matrix of Toeplitz form. Such matrices occur in fourth, or higher, order accurate finite difference approximation to second order elliptic problems, when solving the bi-harmonic problem by a Fourier method, in higher order spline interpolation, etc. We will first consider a corresponding doubly infinite Toeplitz matrix

$$A = \begin{pmatrix}
\ddots & & & & & & & & & & \\
 & \ddots & & & & & & & & & \\
 & & \ddots & & & & & & & & \\
\cdots\ 0 & a_k & a_{k-1} & \cdots & a_0 & a_1 & \cdots & a_{k-1} & a_k & 0 & \cdots \\
\cdots & 0 & a_k & a_{k-1} & \cdots & a_0 & a_1 & \cdots & a_{k-1} & a_k & 0\ \cdots \\
 & & & \ddots & a_1 & a_0 & a_1 & \cdots & & \ddots & \\
 & & & & \ddots & \vdots & \vdots & & & & \\
 & & & & & a_k & \cdots & a_1 & a_0 \cdots & a_k & \\
 & & & & & & \ddots & & \ddots & &
\end{pmatrix}$$

We assume that all a_i are real and that $a_k \neq 0$. To be able to find a LL^T factorization of A, where L is a real lower triangular Toeplitz matrix, we make an assumption analogous to the requirement of positive definiteness for the Cholesky algorithm. Denote by x^* the complex conjugate transpose of the vector x.

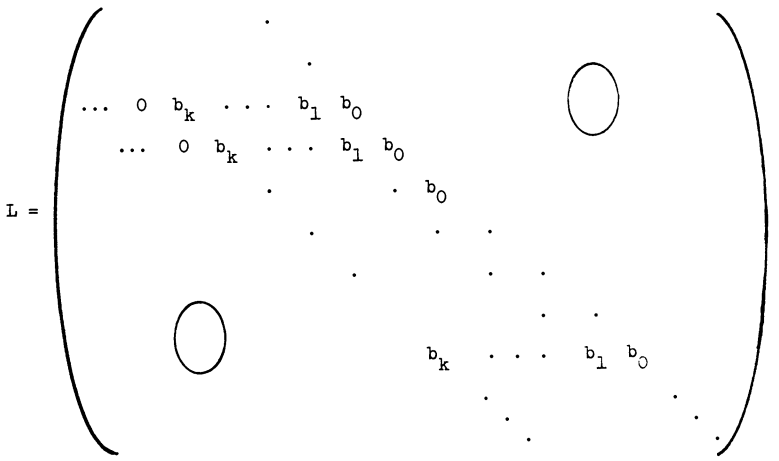
Assumption 1. For all x such that $x^*x < \infty$, $x^*Ax \geqq 0$.

The characteristic function $a(z)$ of A is defined by

$$a(z) = a_k z^k + \cdots + a_0 + \cdots + a_k z^{-k}.$$

We will now prove a lemma, which, in essence, is identical to the Fejér-Riesz lemma; cf. Riesz and Nagy [20, pp. 117–118].

LEMMA 1. *If Assumption* 1 *is satisfied,* $a(z)$ *can be factored as* $a(z) = l(z)\cdot l(1/z)$, *where* $l(z) = b_0 + \cdots + b_k z^k$, $b_0 > 0$, *is a real polynomial with no roots inside the unit circle. Correspondingly, the Toeplitz matrix* A *can be factored as* $A = LL^T$ *where*



Proof. We first factor $a(z) + \epsilon$, $\epsilon > 0$. We note that, if $a(z_0) + \epsilon = 0$, then $a(1/z_0) + \epsilon = a(\bar{z}_0) + \epsilon = a(1/\bar{z}_0) + \epsilon = 0$. The function $a(z) + \epsilon$ has no roots on the unit circle. Assume by contradiction that $a(e^{i\theta}) + \epsilon = 0$ for some real θ. By Assumption 1, $x^*(A + \epsilon I)x \geqq \epsilon x^*x$. We will now reach a contradiction by choosing $x_{(n)} = (1/n)^{1/2}(0, \cdots, 0, 1, e^{i\theta}, e^{2i\theta}, \cdots, e^{i\theta(n-1)}, 0, \cdots)^T$, because $x_{(n)}{}^*x_{(n)} = 1$ while $x_{(n)}{}^*(A + \epsilon I)x_{(n)} \to 0$ when $n \to \infty$. The function $a(z) + \epsilon$ can therefore be factored as $l_\epsilon(z) \cdot l_\epsilon(1/z)$ in such a way that the roots of the real polynomial $l_\epsilon(z)$ lie outside the unit circle. Since the roots of $a(z) + \epsilon = 0$ depend continuously on ϵ, the proof is concluded by letting $\epsilon \to 0$.

A different choice of factors, allowing for roots of $l(z)$ inside the unit circle, would lead to an exponential growth of round-off error; cf. the discussion of the special tri-diagonal case above.

One can also show easily that any function $a(z)$ of the above form, which has no roots of an odd multiplicity on the unit circle, corresponds to a Toeplitz matrix satisfying Assumption 1.

The factors $l(z)$ and $l(1/z)$ of $a(z)$ are known as the Hurwitz factors. They can be computed numerically in different ways. If $k = 1$ or 2, the b_i's can be found by a straightforward approach at the expense of solving one and two quadratic equations, respectively. In the general case, the factors can, of course, be found via the computation of the roots of $a(z) = 0$, but it is more satisfactory to compute $l(z)$ directly. The algorithms to be discussed require strict positive definiteness. As a first step, we therefore consider the possibility of removing the factors corresponding to the roots on the unit circle in order to reduce the problem to a positive definite one. Frequently, it is natural, from the particular context, to make an additional assumption; cf. Thomée [24].

Assumption 2. The Toeplitz matrix A is elliptic if, for some integer m and some positive constant c,

$$a(e^{i\theta}) \geqq c\,|\theta|^{2m} \quad \text{for all } \theta \in [-\pi, \pi].$$

It is easy to see that Assumptions 1 and 2 will allow no zeros on the unit circle except at $z = 1$. The corresponding factors can easily be factored out.

In a general case, when only Assumption 1 is known to hold, one can try Euclid's algorithm to determine if $a(z)$ has any multiple roots. If $z^k a(z)$ and its first derivative have no nontrivial common factor, there are no multiple roots and thus no roots on the unit circle. If a common factor is found, the algorithm can be used to find common factors of $z^k a(z)$ and its successive derivatives, resulting in a factorization of $z^k a(z)$ into lower order polynomials which have only simple roots. This procedure sometimes fails to reduce the problem to strictly positive definite cases. To see this, we can construct a polynomial with double roots on, and several double roots outside and inside, the unit circle. It appears that, in such a case, an iterative root-finding algorithm has to be employed for the approximate calculation of the Hurwitz factors.

We will now discuss an algorithm, suggested by Bauer [3], [4] and others. Its convergence is a Volksatz (folk theorem) among people working with Toeplitz theory; cf. also Rissanen and Barbosa [21]. Let A_+ be a real, semi-infinite symmetric matrix, the rows of which equal those of the Toeplitz matrix A from a certain row onwards. Assume that all principal submatrices of A_+ are strictly positive definite and that the characteristic function $a(z)$, corresponding to A, has no roots on the unit circle. Then, the rows of the lower triangular matrix L_+, normalized to have positive diagonal elements, in the factorization $A_+ = L_+ L_+^T$ will approach those of the Toeplitz matrix L, $A = LL^T$, when the diagonal elements of L are chosen to be positive.

If we apply the algorithm to a tri-diagonal case with $a_1 = -1$ and $a_0 > 2$, it can be shown that the convergence is linear; cf. Bauer [3], [4] and Malcolm and Palmer [18]. In the semidefinite case, $a_0 = 2$, we still have convergence but the error decreases only as $1/n$.

An alternative method, also for strictly positive definite cases, is suggested by Wilson [27]. It is based on the Newton-Raphson method, is quadratically convergent, and is shown to be globally convergent for a family of easily constructible initial approximations of $l(z)$.

We will now assume that the Toeplitz matrix L is available. To illustrate our method, we consider in some detail a case where we want to solve a linear system of equations with n unknowns with a matrix A_n equal to a principal submatrix of the doubly infinite Toeplitz matrix A. Such linear systems arise if we approximate a one-dimensional elliptic problem with Dirichlet boundary conditions by a finite difference approximation of elliptic type (cf. Assumption 2) and prescribe, as boundary conditions, the values of u and differences of u of order one through $k - 1$. This problem leads to a particular choice of the matrices U and V in the Woodbury formula. The modification of our procedure to other matrices, which differ from the case under study in only a few rows can be worked out quite easily. We will always assume that, in our applications, k is considerably much smaller than n.

If we use the relation

$$A = LL^T,$$

we can easily verify that

$$A_n = L_n L_n^T + U_n U_n^T$$

where L_n and A_n are the $n \times n$ Toeplitz matrices

$$L_n = \begin{pmatrix} b_0 & 0 & & & & & & & \\ b_1 & b_0 & & & & & & & \\ \vdots & \vdots & \ddots & & & & & & \\ b_k & b_{k-1} & \cdots & b_0 & & & & & \\ 0 & b_k & \cdots & b_1 & b_0 & & & & \\ & \ddots & \ddots & \vdots & \vdots & \ddots & & & \\ & & \ddots & b_k & b_{k-1} & \cdots & b_0 & & \\ & & & 0 & b_k & \cdots & b_1 & b_0 & \\ & & & & \ddots & \ddots & \vdots & \vdots & \ddots \end{pmatrix}$$

$$A_n = \begin{pmatrix} a_0 & a_1 & \cdots & a_k & 0 & & & & \\ a_1 & a_0 & \cdots & a_{k-1} & a_k & \ddots & & & \\ \vdots & \vdots & \ddots & \vdots & \vdots & \ddots & 0 & & \\ a_k & a_{k-1} & \cdots & a_0 & a_1 & \cdots & a_k & \ddots & \\ 0 & a_k & \ddots & a_1 & a_0 & \ddots & \vdots & \ddots & \\ & \ddots & \ddots & \vdots & \vdots & \ddots & a_{k-1} & \cdots & a_k \\ & & \ddots & 0 & a_k & \cdots & a_1 & a_0 & \cdots \end{pmatrix}$$

The rectangular $n \times k$ matrix U_n is given by

$$U_n = \begin{pmatrix} I_k \\ 0 \end{pmatrix} \tilde{U}$$

where

$$\tilde{U} = \begin{pmatrix} b_k & b_{k-1} & \cdots & & b_1 \\ & b_k & b_{k-1} & \cdots & b_2 \\ & & \ddots & \ddots & \vdots \\ & & & \ddots & \vdots \\ & & & & b_{k-1} \\ & & & & b_k \end{pmatrix}$$

The Woodbury formula then takes the form

$$A_n^{-1} = B_n^{-1} - B_n^{-1} U_n \left(I_k + \tilde{U}^T \begin{pmatrix} I_k \\ 0 \end{pmatrix}^T L_n^{-T} L_n^{-1} \begin{pmatrix} I_k \\ 0 \end{pmatrix} \tilde{U} \right)^{-1} U_n^T B_n^{-1}$$

where $B_n = L_n L_n^T$. To calculate the $k \times k$ matrix

$$C = I_k + \tilde{U}^T \begin{pmatrix} I_k \\ 0 \end{pmatrix}^T L_n^{-T} L_n^{-1} \begin{pmatrix} I_k \\ 0 \end{pmatrix} \tilde{U},$$

we will first solve for the $n \times k$ matrix Y defined by

$$L_n Y = \begin{pmatrix} I_k \\ 0 \end{pmatrix},$$

and, thereafter, form the $k \times k$ matrix $W = Y^T Y$. The matrix C can be expressed in terms of Y and W as

$$C = I_k + \tilde{U}^T Y^T Y \tilde{U} = I_k + \tilde{U}^T W \tilde{U}.$$

The elements of W can be formed quite inexpensively if we take advantage of the Toeplitz structure of L_n. Let $\tilde{y} = (\tilde{y}_1, \cdots, \tilde{y}_n)^T$ be the solution of $L_n \tilde{y} = e^{(1)}$ where $e^{(1)} = (1, 0, \cdots, 0)^T$. The elements y_{ij} of Y are simply given by

$$\begin{aligned} y_{ij} &= \tilde{y}_{i+1-j}, \qquad i \geqq j, \\ &= 0, \qquad\quad i < j. \end{aligned}$$

The element w_{ij} of W is defined by the inner product of the ith and jth columns of Y. Thus

$$w_{ij} = \sum_{\nu=1}^{n+1-i} \tilde{y}_\nu \tilde{y}_{\nu+i-j}, \qquad j = 1, 2, \cdots, i,$$

and, by symmetry, $w_{ij} = w_{ji}$ for $j > i$. For $j \leqq i$, we find from the above formula that

$$w_{ij} = w_{i+1,j+1} + \tilde{y}_{n+1-i} \tilde{y}_{n+1-j}. \tag{3.1}$$

It therefore suffices to compute the last row of W and thereafter find the rest of its elements by formula (3.1) and the symmetry condition. The computation of W will

thus require only $(2k + 1)n + \mathcal{O}(k^2)$ multiplicative and $(2k - 1)n + \mathcal{O}(k^2)$ additive operations. The matrix C can now be found in $\mathcal{O}(k^3)$ operations and it is thereafter factored by using the Cholesky procedure.

We want to point out that the elements w_{kj}, $j = 1, 2, \cdots, k$, can be found by simultaneous accumulation of k inner products while calculating the vector y. Since L_n is a band matrix, we need only $\tilde{y}_{\nu+1-k}, \cdots, \tilde{y}_\nu$ to calculate $\tilde{y}_{\nu+1}$. The whole calculation of W can therefore be organized so that it requires only $k^2 + k + 1$ storage locations.

We will now describe a method for solving $A_n x = f$ which requires very little storage. We write the solution in the form

$$x = A_n^{-1} f = L_n^{-T}(L_n^{-1} f - L_n^{-1} U_n C^{-1} U_n^T L_n^{-T} L_n^{-1} f)$$

and begin by solving $L_n z = f$. Because of the sparseness of U_n, the second vector in the parentheses depends only on the k first components of the vector v given by $L_n^T v = z$. These components can thus be found by back substitution during which only k components of v are carried at any time. The $k \times k$ system $C\tilde{w} = \tilde{U}^T \tilde{v}$, where $\tilde{v} = (I_k \vdots 0)v$, is solved by using the Cholesky decomposition of C. The n-vector $U_n \tilde{w}$ has zeros in the last $n - k$ places. We can therefore consecutively compute the components of $L_n^{-1} U_n \tilde{w}$ while keeping only k previous values of the vector in storage. As soon as a component has been computed, we use it to modify the corresponding component of $L_n^{-1} f$. Finally, we solve $L_n^T x = L_n^{-1} f - L_n^{-1} U_n \tilde{w}$. If we do not wish to retain the data vector f, the whole calculation can be carried out using only $k + 1$ extra storage locations in addition to the k^2 locations needed for the Cholesky decomposition of C.

Our method easily generalizes to a nonsymmetric matrix A of the form $L_n L_n^T + U_n V_n^T$. In our discussion, we assume that $U_n = (I_k \vdots 0)^T$ and that $V_n = (\tilde{V} \vdots 0)^T$ where $\tilde{V}$ is a $k \times k$ matrix. We can then again use the matrix W to compute

$$\begin{aligned} I_k + V_n^T B_n^{-1} U_n &= I_k + \tilde{V}^T \begin{pmatrix} I_k \\ 0 \end{pmatrix}^T L_n^{-T} L_n^{-1} \begin{pmatrix} I_k \\ 0 \end{pmatrix} \\ &= I_k + \tilde{V}^T W. \end{aligned}$$

In the special symmetric case which we have considered, one additional trick improves the algorithm even further. The Woodbury formula can be rewritten as

$$A_n^{-1} = B_n^{-1} - B_n^{-1} \begin{pmatrix} I_k \\ 0 \end{pmatrix}^T (\tilde{U}^{-T} \tilde{U}^{-1} + W)^{-1} \begin{pmatrix} I_k \\ 0 \end{pmatrix} B_n^{-1}. \tag{3.2}$$

The inverse of a triangular Toeplitz matrix is itself a Toeplitz matrix and the last column x of $\tilde{U}^{-1}$ will therefore uniquely define the whole matrix. To find x, we solve the triangular system of linear equations $\tilde{U}x = e^{(k)}$. The elements of $\tilde{U}^{-T}\tilde{U}^{-1}$ can be computed using the same idea as when finding $Y^T Y$. In general cases, the method previously presented provides a more efficient algorithm than the alternative Woodbury formula (3.2).

It is of interest to compare the Toeplitz method with the regular Cholesky factorization. The Cholesky factorization into LDL^T form, $L_{ii} = 1$, requires essentially $k(k + 3)n/2$ multiplications/divisions and $k(k + 1)n/2$ additions/subtractions.

To store the factors, we need $(k + 1)n$ locations and to solve essentially $(2k + 1)n$ multiplications/divisions and $2kn$ additions/subtractions. The major disadvantage of the Toeplitz method is that it requires twice as many operations for solving the system as the Cholesky method. However, in a situation where we cannot retain the Cholesky factors in storage but can afford to save the b_i's and the triangular factors of $I_k + U_n^T(L_nL_n^T)^{-1}U_n$, we find that the Toeplitz method will be more economical in terms of arithmetic operations for $k \geqq 3$. For $k = 2$, the two methods will require the same number of multiplicative operations. The major advantage of the Toeplitz method is that we will never need more than $n + \mathcal{O}(k^2)$ storage locations if we do not wish to retain the data vector. As pointed out above, our method can also be used, with equal economy for nonsymmetric perturbations of the matrix $L_nL_n^T$. In such cases, the Cholesky method is no longer applicable. One can show that, under the same assumptions as in the above comparison of the Cholesky and Toeplitz methods, our algorithm is preferable in terms of arithmetic operations to the regular band Gaussian elimination procedure already for $k = 2$.

Rose [22] explored the use of a method similar to ours in a tri-diagonal case where what corresponds to our doubly infinite Toeplitz matrix A is the product of two Toeplitz matrices and a diagonal nonconstant matrix. It is obvious that most of our considerations are valid in cases where there is a convenient doubly infinite matrix A for which Toeplitz and diagonal factors can be found. That is, for example, frequently possible for standard difference approximations to operators of the form $-\partial_x a(x)\partial_x$.

4. Multi-Dimensional Problems. We will now discuss the use of Toeplitz methods for matrices of block-band form which arise from finite difference approximations of separable elliptic problems. We first consider the standard five-point finite difference approximation of Poisson's equation on a rectangular region with Dirichlet boundary conditions. The matrix, which is block tri-diagonal, has the form

$$A = \begin{pmatrix} A_0 & -I & & & & \\ -I & A_0 & -I & & & \\ & -I & A_0 & \cdot & & \\ & & \cdot & \cdot & \cdot & \\ & & & \cdot & \cdot & \cdot \\ & & & & \cdot & \cdot & -I \\ & & & & & -I & A_0 \end{pmatrix}$$

where A_0 is the tri-diagonal, $n \times n$ matrix

$$A_0 = \begin{pmatrix} 4 & -1 & & & & & \\ -1 & 4 & -1 & & & & \\ & -1 & 4 & \cdot & & & \\ & & \cdot & \cdot & \cdot & & \\ & & & \cdot & \cdot & \cdot & \\ & & & & \cdot & \cdot & -1 \\ & & & & & -1 & 4 \end{pmatrix}$$

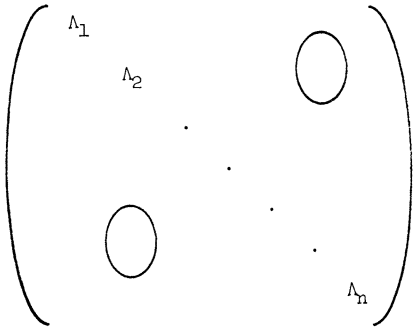
If we attempt to find a factorization of A, or of some lower rank perturbation of A, corresponding to those of Section 3, we must find an appropriate factorization of the characteristic function $a(z_1, z_2) = -z_1 - z_2 + 4 - z_1^{-1} - z_2^{-1}$. When the characteristic function depends on several variables, a factorization like the one of Lemma 1 is possible only in exceptional cases. This fact can be expressed by saying that the Fejér-Riesz theorem does not extend to several variables or, alternatively, that no LL^T factorization is possible of the infinite matrix corresponding to $a(z_1, z_2)$ such that L has only a fixed number of nonzero elements in each row. It is, for example, not difficult to show that the above characteristic function $a(z_1, z_2)$ cannot be factored in a useful way. We therefore turn our attention to the separation of variables technique. The normalized eigenvectors $\varphi^{(l)}$, $l = 1, 2, \cdots, n$, of A_0 are given by $\varphi_j^{(l)} = (2/(n+1))^{1/2} \sin(jl\pi/(n+1))$, $j = 1, \cdots, n$. The orthonormal matrix Q, with the eigenvectors $\varphi^{(l)}$ for columns, satisfies $Q^T A_0 Q = D_0$, where D_0 is the diagonal matrix of eigenvalues of A_0, $\lambda_l = (D_0)_{ll} = 4 - 2\cos(l\pi/(n+1))$.

The change of basis which corresponds to the diagonalization of A_0 can be carried out inexpensively by using the fast Fourier transform (FFT) (cf. Cooley, Lewis and Welch [9]) if $n + 1$ has many prime factors, and, in particular, if $n + 1$ is a power of 2. After this change of basis, the matrix A transforms into

$$\begin{pmatrix} Q^T & & & & \\ & Q^T & & & \\ & & \cdot & & \\ & & & \cdot & \\ & & & & Q^T \end{pmatrix} \begin{pmatrix} A_0 & -I & & & & & \\ -I & A_0 & -I & & & & \\ & \cdot & \cdot & \cdot & & & \\ & & \cdot & \cdot & \cdot & & \\ & & & \cdot & \cdot & \cdot & \\ & & & & \cdot & \cdot & -I \\ & & & & & -I & A_0 \end{pmatrix} \begin{pmatrix} Q & & & & \\ & Q & & & \\ & & \cdot & & \\ & & & \cdot & \\ & & & & Q \end{pmatrix}$$

$$= \begin{pmatrix} D_0 & -I & & & & & \\ -I & D_0 & -I & & & & \\ & \cdot & \cdot & \cdot & & & \\ & & \cdot & \cdot & \cdot & & \\ & & & \cdot & \cdot & \cdot & \\ & & & & \cdot & \cdot & -I \\ & & & & & -I & D_0 \end{pmatrix}$$

A linear system of equations with this coefficient matrix can be solved easily by the Toeplitz method developed in Section 2, because the permutation of rows and columns which preserves symmetry and groups the lth equation of every block together into one block, leads to a direct sum of Toeplitz matrices



where

$$\Lambda_l = \begin{pmatrix} \lambda_l & -1 & & & & & \\ -1 & \lambda_l & -1 & & & & \\ & -1 & \ddots & \ddots & & & \\ & & \ddots & \ddots & \ddots & & \\ & & & \ddots & \ddots & \ddots & \\ & & & & \ddots & \ddots & -1 \\ & & & & & -1 & \lambda_l \end{pmatrix}$$

The algorithm is thus carried out in three steps. First, we apply the FFT variant which corresponds to a real sine transform, to the blocks of the appropriately partitioned data vector. The transformed vector is then permuted and the n tri-diagonal systems are solved by the Toeplitz method. Finally, an inverse Fourier transform is applied, after a permutation, to the blocks of the partitioned vector.

We remark that the FFT involves a certain permutation corresponding to the inverse binary ordering, and that the inverse FFT involves the transpose of this permutation; cf. Cooley, Lewis and Welch [9]. These permutations can be eliminated from the algorithm if the block matrices Λ_l are permuted suitably.

We next consider the same finite difference approximations, but with the Dirichlet condition replaced by a periodicity condition in both directions. The corresponding matrix C takes the form

$$C = \begin{pmatrix} C_0 & -I & 0 & & & 0 & -I \\ -I & C_0 & -I & & & & 0 \\ 0 & \ddots & \ddots & \ddots & & & \\ & & \ddots & \ddots & \ddots & & \\ & & & \ddots & \ddots & \ddots & 0 \\ & & & & \ddots & \ddots & -I \\ -I & 0 & & & 0 & -I & C_0 \end{pmatrix}$$

where C_0 is the $n \times n$ matrix

$$C_0 = \begin{pmatrix} 4 & -1 & 0 & \cdots & 0 & -1 \\ -1 & 4 & -1 & & & 0 \\ 0 & \ddots & \ddots & \ddots & & \vdots \\ \vdots & & \ddots & \ddots & \ddots & \\ & & & \ddots & \ddots & 0 \\ 0 & & & & -1 & \\ -1 & 0 & \cdots & 0 & -1 & 4 \end{pmatrix}$$

For convenience, we assume that n is an even number. The matrix C_0 has the normalized eigenvectors $(1/n)^{1/2}(1, 1, \cdots, 1)^T$ and $(1/n)^{1/2}(1, -1, \cdots, -1)$ corresponding to the simple eigenvalues 2 and 6, respectively, and $(n - 2)/2$ double eigenvalues $4 - 2\cos(2\pi l/n)$, $l = 1, \cdots, (n - 2)/2$, with the corresponding eigenvectors $\varphi_{\mathrm{I}}{}^{(l)}$ and $\varphi_{\mathrm{II}}{}^{(l)}$ given by

$$\varphi_{\mathrm{I},j}^{(l)} = (2/n)^{1/2} \sin(jl2\pi/n), \qquad \varphi_{\mathrm{II},j}^{(l)} = (2/n)^{1/2} \cos(jl2\pi/n).$$

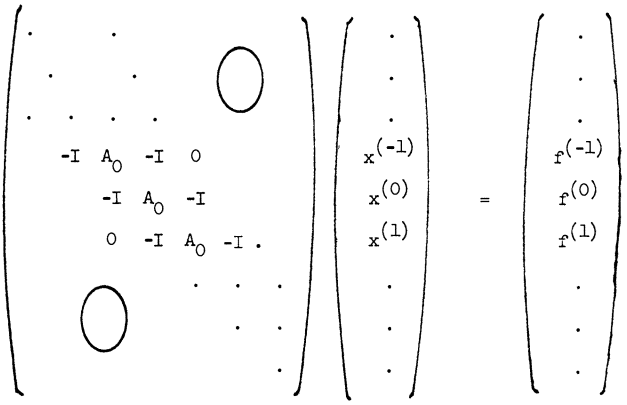
The corresponding change of basis can again be executed with the aid of the FFT. The tri-diagonal Toeplitz matrices of the Dirichlet case are now replaced by the matrices

$$\Gamma_l = \begin{pmatrix} \gamma_l & -1 & 0 & \cdots & 0 & -1 \\ -1 & \gamma_l & -1 & & & 0 \\ 0 & -1 & \ddots & \ddots & & \vdots \\ \vdots & & \ddots & \ddots & \ddots & \\ & & & \ddots & \ddots & 0 \\ 0 & & & & -1 & \\ -1 & 0 & \cdots & 0 & -1 & \gamma_l \end{pmatrix}$$

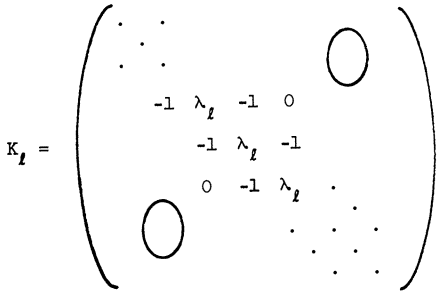
where $\gamma_l = 4 - 2\cos(2\pi l/n)$, $l = 0, 1, \cdots, n/2$. The matrix Γ_0 has a simple eigenvalue equal to zero reflecting the singularity of the matrix C. Since $Ce = 0$, where $e = (1, 1, \cdots, 1)^T$, it follows that $Cx = f$ has a solution only if $e^T f = 0$, i.e., the sum of the components of f is zero. The components of the right-hand side $\hat{f}_0$ of the linear system $\Gamma_0 \hat{x}_0 = \hat{f}_0$, which is derived in a way completely analogous to the Dirichlet case, are the Fourier coefficients corresponding to the vector $(1, 1, \cdots, 1)^T$ and the sum of these components will thus vanish if $Cx = f$ is solvable. If we set the last component of $\hat{x}_0$ equal to zero and remove the last equation, the system $\Gamma_0 \hat{x}_0 = \hat{f}_0$ reduces to a tri-diagonal, nonsingular system of equations of Toeplitz form. Its solution, augmented with zero, can be shown, by the linear dependence of the equations, to satisfy the original system. The remaining linear systems of equations, with $l \geqq 1$, can be solved by the Toeplitz method modifying the (1, 1), (1, n) and (n, 1) elements of Γ_l. After the inverse Fourier transform step, this algorithm will have produced a particular solution of $Cx = f$. Any other solution to the problem will differ from this particular solution by a constant.

As a third example, we consider the solution of the five-point difference ap-

proximation of Poisson's equation $-\Delta u = f$ on an infinite parallel strip, $-\infty < x < \infty$, $0 \leqq y \leqq 1$. We impose homogeneous Dirichlet conditions at $y = 0$ and $y = 1$ and assume that f vanishes outside a bounded region. The problem takes the form



where $x^{(i)}$ denotes the vector of values of the approximate solution for $x = ih$ and h denotes the mesh length in the x-direction. The matrix A_0 is defined as in our first example. It is now natural to use a Fourier transform with respect to y. By using the same variant of the FFT as for the Dirichlet problem discussed above, we reduce our problem to the solution of n linear systems of equations. These systems are of infinite order with tri-diagonal coefficient matrices K_l of Toeplitz form



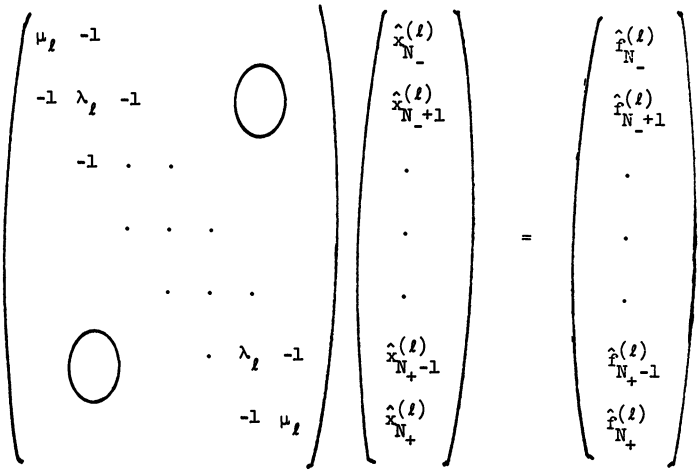
where, as before, $\lambda_l = 4 - 2\cos(l\pi/(n+1))$, $l = 1, 2, \cdots, n$. From our assumption on the function f, we see that the components $\hat{f}_i^{(l)}$ of the data vector $\hat{f}^{(l)}$ in the system $K_l\hat{x}^{(l)} = \hat{f}^{(l)}$ will vanish for $j < N_-$ and $j > N_+$ if N_- and N_+ are chosen large enough. We can therefore write

$$-\hat{x}_{j-1}^{(l)} + \lambda_l\hat{x}_j^{(l)} - \hat{x}_{j+1}^{(l)} = 0, \quad \text{for } j < N_- \text{ and } j > N_+.$$

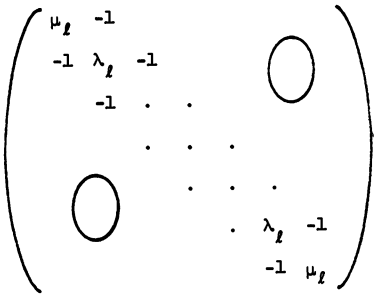
This homogeneous difference equation has the solution const $\mu_l^{\,j}$ + const $\mu_l^{\,-j}$ where $\mu_l = \lambda_l/2 + (\lambda_l^{\,2}/4 - 1)^{1/2} > 1$. Imposing the obvious free space boundary conditions, i.e., requiring that the solution remains bounded for all n, we find that

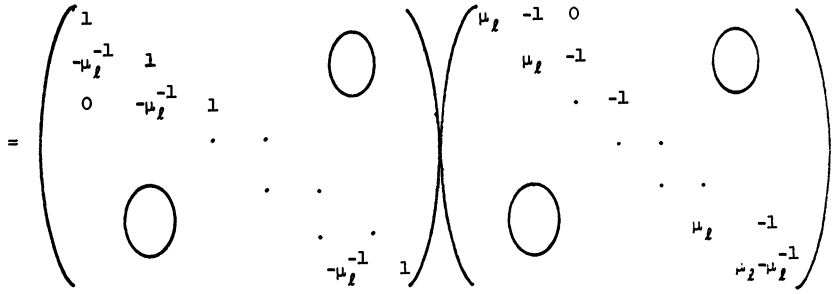
$$\begin{aligned} \hat{x}_j^{(l)} &= \hat{x}_{N_+}^{(l)} \cdot \mu_l^{(N_+ - j)}, \qquad j \geqq N_+, \\ \hat{x}_j^{(l)} &= \hat{x}_{N_-}^{(l)} \cdot \mu_l^{(j - N_-)}, \qquad j \leqq N_-. \end{aligned} \tag{4.1}$$

By using relation (4.1) for $j = N_+ + 1$ and $j = N_- - 1$ and some elementary algebra, we can set up a finite linear system of equations:



This system can be solved very nicely by the Toeplitz method because

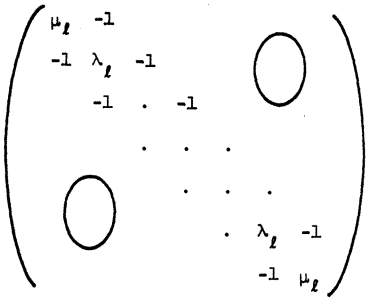


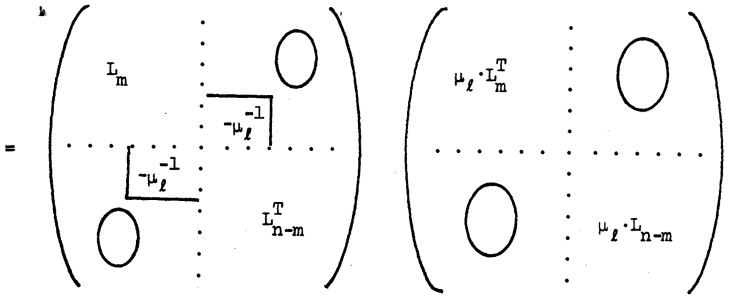


Finally, the values of the solution on any mesh line parallel to the y-axis can be found via an inverse Fourier transform.

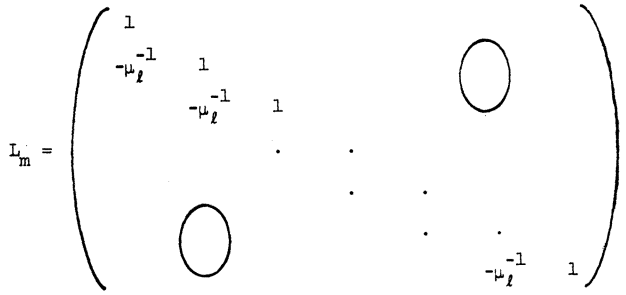
A half infinite strip, $0 \leqq x < \infty$, $0 \leqq y \leqq 1$, with the Dirichlet boundary condition added at $x = 0$, $0 \leqq y \leqq 1$, can also be handled easily by the Fourier-Toeplitz method without using the Woodbury formula. Only a change of the right-most element of the last row of the coefficient matrices and their upper triangular factors have to be made in comparison with the case discussed above, if we order the unknowns from right to left.

For further discussion of the solution of Poisson's equation on infinite strips, cf. Buneman [6]. Buneman has also suggested the following alternative method of solving the linear system of equations just considered. We refer to it as a twisted Toeplitz factorization; for a related idea, cf. Strang [23]. The matrix under study can, as is easily verified, be written as a product of two, very special tri-diagonal matrices





where the $m \times m$ matrix L_m is given by



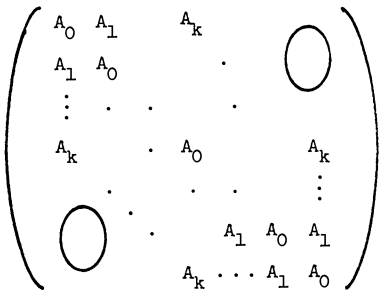
The solution of the linear system of equations corresponding to the first matrix reduces to an inward sweep and the solution of a 2×2 system of the form

$$\begin{bmatrix} 1 & -\mu_l^{-1} \\ -\mu_l^{-1} & 1 \end{bmatrix} \begin{bmatrix} y_m \\ y_{m+1} \end{bmatrix} = \begin{bmatrix} f_m + \mu_l^{-1} y_{m-1} \\ f_{m+1} + \mu_l^{-1} y_{m+2} \end{bmatrix}.$$

The second matrix corresponds to a simple outward sweep. This algorithm will not save arithmetic operations compared to the previous one, but it has a nice symmetry, in that it treats the two endpoints in the same way.

Twisted Toeplitz factorizations can be worked out for general Toeplitz matrices but, in the general case, their use must of course be combined with the Woodbury formula for necessary modifications of certain matrix elements. It does not seem to offer any particular computational advantages to the methods discussed in Section 3.

The Fourier-Toeplitz method can be extended to positive definite, symmetric matrices of the form



where all the block matrices A_i commute and can be simultaneously diagonalized by a change of variables corresponding to one of the FFT variants. Certain finite difference approximations to the bi-harmonic equation with boundary conditions which allow for separation of variables belong to this category as well as certain fourth order finite difference approximations to Poisson's equation.

A possible improvement of the Fourier-Toeplitz method, which has not been tried experimentally, could be obtained by one or a few block odd-even reduction steps before the FFT is applied; cf. Hockney [15], [16], Buzbee, Golub, and Nielson [7] or Widlund [25]. This would result in the application of the Fourier transform to vectors with fewer components at the expense of an increased band width of the Toeplitz matrices and the work connected with the odd-even reduction steps. This idea has proven quite useful in Hockney's method. Our method can also be extended to three dimensions, if we use FFT for two of the variables.

5. Numerical Results. The Fourier-Toeplitz methods discussed in Section 4 have been tried on the CDC 6600 of the AEC Computing Center at the Courant Institute, New York, and on the IBM 360/75 of the Institute of Technology (K.T.H.) in Stockholm. We will describe the results of some of our tests in New York. The programs were written in assembly language and the execution time of one of them was compared with a machine code program implementing Buneman's algorithm which Dr. B. Buzbee of Los Alamos was kind enough to make available to us. The FFT program used was kindly made available to us by Dr. R. Singleton of Stanford Research Institute. The rounding errors only affected the last few digits in all our experiments.

Case 1. Poisson's equation with homogeneous Dirichlet boundary conditions.

(a) 127 × 127 mesh. Total time: 1.40 sec. of which 1.12 sec. were used for the Fourier transforms. For the Buneman algorithm: 1.81 sec.

(b) 63 × 63 mesh. Total time: 0.346 sec. For the Buneman algorithm: 0.42 sec. Storage used (excluding the program) $n^2 + 5n$ for an $n \times n$ mesh.

Case 2. Homogeneous Neumann conditions.

(a) 129 × 129 mesh. Total time: 1.57 sec.

(b) 65×65 mesh. Total time: 0.387 sec. Storage used: $n^2 + 7n$.

Case 3. Periodic boundary conditions.

(a) 128×128 mesh. Total time: 1.14 sec. Total time for the Buneman program: 1.90 sec.

(b) 64×64 mesh. Total time: 0.268 sec. Total time for the Buneman program: 0.389 sec. Storage used: $n^2 + 4n$.

Case 4. Infinite strip with Dirichlet boundary condition.

127×127 mesh, 127 mesh points in the x-direction were involved in the solution of the tri-diagonal linear systems of equations and in the inverse Fourier transform steps. Total time: 1.42 sec. Storage used: $n^2 + 4n$.

Another set of numerical experiments was carried out at Uppsala University in order to test the numerical stability of the algorithms discussed in Section 3. Two Algol 60 programs, Toeplitz I and II, which both implement Eq. (3.2), were run, in double precision, on an IBM 370/155. The first algorithm uses the modification technique which we developed in Section 3 in order to save storage. In the second algorithm, the data vector is retained and its first k components altered after the solution of the $k \times k$ linear system of equations. The performance of our algorithms was compared with that of Martin and Wilkinson's Cholesky program; cf. Wilkinson and Reinsch [26, p. 50].

Case 1. A linear system of equations with the 100×100 principal submatrix of the doubly infinite Toeplitz matrix corresponding to the characteristic function

$$\begin{aligned} a(z) &= z^2 - 4z + 6 - 4z^{-1} + z^{-2} \\ &= (1 - 2z + z^2)(1 - 2z^{-1} + z^{-2}). \end{aligned}$$

This problem is quite ill conditioned. The relative error in maximum norm:

Toeplitz I: 2.5×10^{-8}
Toeplitz II: 6.0×10^{-10}
Cholesky: 5.5×10^{-11}

Case 2. A quite well-conditioned problem corresponding to

$$\begin{aligned} a(z) &= 4z^2 + 5z + 18 + 5z^{-1} + 4z^{-2} \\ &= (4 + z + z^2)(4 + z^{-1} + z^{-2}). \end{aligned}$$

The three algorithms produced solutions differing from the correct one only in the last digit.

Courant Institute of Mathematical Sciences
New York University
251 Mercer Street
New York, New York 10012*

Computer Science Department
Stanford University
Stanford, California 94305

Computer Science Department
Uppsala University
Sturegatan 4B
Uppsala, Sweden

* Address of first, fourth and fifth authors.

1. M. D. BAKES, "An alternative method of solution of certain tri-diagonal systems of linear equations," *Comput. J.*, v. 7, 1964, pp. 135–136. MR **31** #6353.

2. E. H. BAREISS, "Numerical solution of linear equations with Toeplitz and vector Toeplitz matrices," *Numer. Math.*, v. 13, 1969, pp. 404–424. MR **40** #8234.

3. F. L. BAUER, "Ein direktes Iterationsverfahren zur Hurwitz-Zerlegung eines Polynoms," *Arch. Elec. Ubertr.*, v. 9, 1955, pp. 285–290. MR **17,** 900.

4. F. L BAUER, "Beiträge zur Entwicklung numerischer Verfahren für programmgesteuerte Rechenanlagen. II. Direkte Faktorisierung eines Polynoms," *Bayer. Akad. Wiss. Math.-Nat. Kl. S.-B.*, v. 1956, pp. 163–203. MR **19,**686.

5. O. BUNEMAN, *A Compact Non-Iterative Poisson Solver*, Rep. SUIPR-294, Inst. Plasma Research, Stanford University, 1969.

6. O. BUNEMAN, *Inversion of the Helmholtz (or Laplace-Poisson) Operator in Slab Geometry*, Rep. SUIPR-467, Inst. Plasma Research, Stanford University, 1972.

7. B. L. BUZBEE, G. H. GOLUB & C. W. NIELSON, "On direct methods for solving Poisson's equation," *SIAM J. Numer. Anal.*, v. 7, 1970, pp. 627–656. MR **44** #4920.

8. B. L. BUZBEE, F. W. DORR, J. A. GEORGE & G. H. GOLUB, "The direct solution of the discrete Poisson equation on irregular regions," *J. SIAM Numer. Anal.*, v. 8, 1971, pp. 722–736. MR **45** #1403.

9. J. W. COOLEY, P. A. W. LEWIS & P. D. WELCH, "The fast Fourier transform algorithm: Programming consideration in the calculation of sine, cosine and Laplace transform," *J. Sound Vib.*, v. 12, 1970, pp. 315–337.

10. D. J. EVANS, "An algorithm for the solution of certain tri-diagonal systems of linear equations," *Comput. J.*, v. 15, 1972, pp. 356–359.

11. D. J. EVANS & C. V. D. FORRINGTON, "Note on the solution of certain tri-diagonal systems of linear equations," *Comput. J.*, v. 5, 1962/63, pp. 327–328. MR **27** #6377.

12. G. E. FORSYTHE & C. B. MOLER, *Computer Solution of Linear Algebraic Systems*, Prentice-Hall, Englewood Cliffs, N.J., 1967. MR **36** #2306.

13. J. A. GEORGE, "Block elimination of finite element systems of equations," *Sparse Matrices and Their Applications*, edited by D. J. Rose and R. A. Willoughby, Plenum Press, New York, 1972.

14. J. A. GEORGE, *The Use of Direct Methods for the Solution of the Discrete Poisson Equation on Non-Rectangular Regions*, Computer Science Report 159, Stanford University, 1970.

15. R. W. HOCKNEY, "A fast direct solution of Poisson's equation using Fourier analysis," *J. Assoc. Comput. Mach.*, v. 12, 1965, pp. 95–113. MR **35** #3913.

16. R. W. HOCKNEY, "The potential calculation and some applications," *Methods in Computational Physics*. Vol. 9, Academic Press, New York, 1970.

17. A. S. HOUSEHOLDER, *The Theory of Matrices in Numerical Analysis*, Blaisdell, New York, 1964. MR **30** #5475.

18. M. MALCOLM & J. PALMER, *A Fast Method for Solving a Class of Tri-Diagonal Linear Systems*, Computer Science Report 323, Stanford University, 1972.

19. W. PROSKUROWSKI & O. B. WIDLUND. (To appear.)

20. F. RIESZ & B. SZ.-NAGY, *Leçons d'analyse fonctionnelle*, 2nd ed., Akad. Kiadó, Budapest, 1953; English transl., *Functional Analysis*, Ungar, New York, 1955. MR **15,**132; **17,**175.

21. J. RISSANEN & L. BARBOSA, "Properties of infinite covariance matrices and stability of optimum predictors," *Information Science*, v. 1, 1968/69, pp. 221–236. MR **39** #5032.

22. D. J. ROSE, "An algorithm for solving a special class of tri-diagonal systems of linear equations," *Comm. ACM*, v. 12, 1969, pp. 234–236. MR **43** #5706.

23. G. STRANG, "Implicit difference methods for initial-boundary value problems," *J. Math. Anal. Appl.*, v. 16, 1966, pp. 188–198. MR **34** #5323.

24. V. THOMÉE, "Elliptic difference operators and Dirichlet's problem," *Contributions to Differential Equations*, v. 3, 1964, pp. 301–324. MR **29** #746.

25. O. B. WIDLUND, "On the use of fast methods for separable finite difference equations for the solution of general elliptic problems," *Sparse Matrices and Their Applications*, edited by D. J. Rose and R. A. Willoughby, Plenum Press, New York, 1972.

26. J. H. WILKINSON & C. REINSCH, "Linear algebra," *Handbook for Automatic Computation*, Springer-Verlag, Berlin and New York, 1971.

27. G. WILSON, "Factorization of the covariance generating function of a pure moving average process," *SIAM J. Numer. Anal.*, v. 6, 1969, pp. 1–7. MR **40** #6775.