A VARIANT OF THE GOHBERG-SEMENCUL FORMULA INVOLVING CIRCULANT MATRICES

By

Gregory Ammar

and

Paul Gader

IMA Preprint Series # 471

December 1988

A Variant of the Gohberg-Semencul Formula Involving Circulant Matrices

Gregory Ammar†
Department of Mathematical Sciences
Northern Illinois University
DeKalb, IL 60115

Paul Gader*
Department of Mathematics
University of Wisconsin-Oshkosh
Oshkosh, WI 54901

Abstract. The Gohberg-Semencul formula expresses the inverse of a Toeplitz matrix as a sum of products of lower triangular and upper triangular Toeplitz matrices. In this paper the idea of cyclic displacement structure is used to show that the upper triangular factors can be replaced by circulant matrices. The resulting computational savings afforded by this modified formula is discussed.

AMS (MOS) Subject classifications: 65F05, 15A09, 15A23

Key words and phrases: Toeplitz matrix, circulant matrix, Gohberg-Semencul formula, displacement, cyclic displacement, fast Fourier transform.

Runniing Title: A Variant of the Gohberg-Semencul Formula

---

† Research of this author was supported in part by the National Science Foundation under grant DMS-8704196.

* Research of this author was supported in part by the Institute for Mathematics and Its Applications with funds provided by the National Science Foundation.

1. Introduction. Let $M = [\mu_{j-k}]_{j,k=0}^{n-1}$ be a real symmetric positive definite Toeplitz matrix of order n. There are several well-known $O(n^2)$ algorithms for solving the linear system of equations $Mx = b$, and more recently several $O(n \log^2 n)$ algorithms have been developed. See, for example, [11],[9],[8],[6],[1],[2] and the references contained therein. Algorithms from both of these classes often rely, either implicitly of explicitly, on the Gohberg-Semencul formula, which provides a factorization of M^{-1} into the sum of products of lower triangular and upper triangular Toeplitz matrices.

Let $e_0, e_1, \ldots, e_{n-1}$ be the columns of the identity matrix I of order n. Since M is positive definite, the last column of M^{-1} can be written as $M^{-1} e_{n-1} = \frac{1}{\delta_{n-1}} r$, where $\delta_{n-1} > 0$ and $r = [\rho_j]_{j=0}^{n-1}$ with $\rho_{n-1} = 1$. The *Gohberg-Semencul formula* is then given by

$$\delta_{n-1} M^{-1} = L(r_1) L(r_1)^T - L(r_0) L(r_0)^T, \tag{1}$$

where $r_0 = Z_n r$, $r_1 = [\rho_{n-j-1}]_0^{n-1}$, $L(r)$ denotes the lower triangular Toeplitz matrix whose first column is r, and $Z_n = L(e_1)$ is the *downshift matrix*. See, for example, [5], [7].

We note in passing that the polynomial $\chi_{n-1}(\lambda) = \sum_{j=0}^{n-1} \rho_j \lambda^j$ is the monic *Szegö polynomial* of degree $n-1$ determined by M, and δ_{n-1} is the norm of χ_{n-1} in the inner product determined by M. Furthermore, the Gohberg-Semencul formula is a manifestation of the Christoffel-Darboux formula for Szegö polynomials [11], [10], [1].

Many algorithms for the solution of a Toeplitz system consist of two phases:

1. the computation of the coefficients of the Szegö polynomial $\chi_{n-1}(\lambda)$ and its squared norm δ_{n-1};

2. the computation of $M^{-1} b$ using the Gohberg-Semencul formula, implemented in $O(n \log n)$ operations using fast Fourier transforms.

The notion of displacement structure underlies many techniques for solving Toeplitz systems of equations, and can be used to extend algorithms for Toeplitz matrices to other classes of matrices [9], [3]. In fact, every square matrix can be written as the sum of products of upper triangular and lower triangular Toeplitz matrices. Furthermore, the number of terms in this sum is small if the matrix is "close" to a Toeplitz matrix in the sense that its *displacement rank* is small. Matrices of small displacement rank can then be treated using extended versions of algorithms for Toeplitz matrices.

The notion of displacement structure has been generalized to include circulant matrices and other group matrices in [4]. In Section 2 we show how the circulant displacement representation of the inverse of a Toeplitz matrix can be used to derive the following factorization of the inverse of the positive definite Toeplitz matrix M.

Proposition 1.

$$\delta_{n-1} M^{-1} = L(r_1)\, C(r_1)^T - L(r_0)\, C(r_1) \tag{2}$$

where $C(r)$ denotes the circulant matrix whose first column is r.

In Section 3 we discuss the computational savings resulting from the use of this formula. We obtain a computational savings of more than 35% over the Gohberg-Semencul formula in obtaining $M^{-1} b$ when n is a power of two.

2. A Circulant Gohberg-Semencul Formula. In the following all matrices are assumed to be real and $n \times n$. If the *displacement* of a square matrix A is given by the sum of α outer products,

$$A - Z A Z^T = \sum_{m=1}^{\alpha} x_m\, y_m^T$$

where $x_m, y_m \in \mathbf{R}^n$, then

$$A = \sum_{m=1}^{\alpha} L(x_m)\, L(y_m)^T$$

This is the *displacement representation* of A developed in [3,9]. The usefulness of this representation for Toeplitz matrices stems from the fact that a Toeplitz matrix and its inverse have displacement rank $\alpha \leq 2$. In particular, the Gohberg-Semencul formula follows from the fact that

$$M^{-1} - Z M^{-1} Z^T = \frac{1}{\delta_{n-1}}(r_1 r_1^T - r_0 r_0^T) .$$

The circulant analog of displacement structure is based on replacing the downshift matrix Z above with the *cyclic downshift matrix* $E = C(e_1)$. In particular, we will need the following result from [4].

Proposition 2. *If the cyclic displacement of* A *is given as the sum*

$$A - E A E^T = \sum_{m=1}^{\alpha} x_m y_m^T , \tag{3}$$

then

$$A = C_l + \sum_{m=1}^{\alpha} L(x_m) C(y_m)^T \tag{4}$$

where C_l *is the circulant matrix with the same last row as that of* A.

Given (3) and (4), the derivation of the corresponding analog of the Gohberg-Semencul formula is straightforward. Let M be a real positive definite Toeplitz matrix and $A = \delta_{n-1} M^{-1}$. Then the cyclic displacement of A is given by

$$\begin{aligned} A - E A E^T &= A - Z A Z^T - e_0 r_0^T - r_0 e_0^T - e_0 e_0^T \\ &= r_1 r_1^T - r_0 r_0^T - e_0 (e_0 + r_0)^T - r_0 e_0^T \end{aligned}$$

since $E = Z + e_0 e_{n-1}^T$ and $Z A e_{n-1} = r_0$. Proposition 2 then yields

$$A = L(r_1) C(r_1)^T - L(r_0) C(r_0)^T - C(r_1) - L(r_0) + C(r_1)$$

because $C(e_0 + r_0)^T = C(r_1)$ and $L(e_0) = C(e_0) = I$. Hence,

$$A = L(r_1)\,C(r_1)^T - L(r_0)(C(r_0)+I)^T$$

$$= L(r_1)\,C(r_1)^T - L(r_0)\,C(r_1) \qquad (2)$$

which is the desired formula.

3. Computational Implications. We now show how the analog of the Gohberg-Semencul formula derived in Section 2 leads to a more efficient way to calculate $M^{-1}b$. The increased computational efficiency is due to the fact that multiplication by a circulant matrix is roughly twice as fast as multiplication by a triangular Toeplitz matrix of the same size. In fact, the efficient multiplication of a Toeplitz matrix and a vector is achieved by embedding the Toeplitz matrix in a circulant matrix of twice the size.

We first recall some fundamental facts regarding fast Fourier transforms and efficient circulant-vector multiplication. Let $\omega_n = e^{2\pi i/n}$ denote the principal nth root of unity. The discrete Fourier transform (DFT) of the n-vector x is defined by $y = F_n\,x$, where $n\,F_n = [\omega_n^{-jk}]_{j,k=0}^{n-1}$, and inverse discrete Fourier transform (IDFT) of y is $x = W_n\,y$, where $W_n \equiv F_n^{-1} = [\omega_n^{jk}]_0^{n-1} = n\,\bar{F}_n$. The computation of $F_n\,x$ and $W_n\,x$ can be performed in $O(n\log n)$ arithmetic operations using any one of many well-known techniques, collectively called fast Fourier transforms (FFTs). Let $\tau(n)$ denote the amount of computation required to perform one real FFT of order n.

Recall that the circulant-vector product $z = C(x)\,y$ is equal to the cyclic convolution of the vectors x and y, which we denote by $x * y$. Moreover, $z = x * y$ if and only if $F_n\,z = (F_n\,x)\cdot(F_n\,y)$, where $x \cdot y$ denotes the componentwise product of x and y. Consequently, $z = W_n((F_n\,x)\cdot(F_n\,y))$, so z can be computed in $3\,\tau(n) + O(n)$ arithmetic operations.

Let us now write the Gohberg-Semencul formula (1) as

$$\delta_{n-1}\,M^{-1} = A = T_1^T\,T_1 - T_0\,T_0^T\ ,$$

where $T_1^T = L(r_1)$ and $T_0 = L(r_0)$. Let $u = T_0^T b$, $v = T_1 b$, $r = T_0 u$ and $s = T_1^T v$. Then $A b = s - r$. Note that

$$\begin{bmatrix} T_0 & T_1 \\ T_1 & T_0 \end{bmatrix} = C\left(\begin{bmatrix} r_0 \\ e_0 \end{bmatrix}\right), \quad \begin{bmatrix} T_0^T & T_1^T \\ T_1^T & T_0^T \end{bmatrix} = C\left(\begin{bmatrix} 0 \\ r_1 \end{bmatrix}\right)$$

are circulant matrices of order $2n$. The following convolution formulas can therefore be used to calculate r and s. The symbols $\times$ denote n-vectors that are irrelevant in the computation.

$$\begin{bmatrix} u \\ \times \end{bmatrix} := \begin{bmatrix} 0 \\ r_1 \end{bmatrix} * \begin{bmatrix} b \\ 0 \end{bmatrix}, \quad \begin{bmatrix} \times \\ v \end{bmatrix} := \begin{bmatrix} r_0 \\ e_0 \end{bmatrix} * \begin{bmatrix} b \\ 0 \end{bmatrix},$$

$$\begin{bmatrix} r \\ \times \end{bmatrix} := \begin{bmatrix} r_0 \\ e_0 \end{bmatrix} * \begin{bmatrix} u \\ 0 \end{bmatrix}, \quad \begin{bmatrix} s \\ \times \end{bmatrix} := \begin{bmatrix} 0 \\ r_1 \end{bmatrix} * \begin{bmatrix} 0 \\ v \end{bmatrix}.$$

In terms of FFTs, the computations can be performed as follows.

$$t := F_{2n} \begin{bmatrix} b \\ 0 \end{bmatrix}, \quad p := F_{2n} \begin{bmatrix} r_0 \\ e_0 \end{bmatrix}, \quad q := F_{2n} \begin{bmatrix} 0 \\ r_1 \end{bmatrix}$$

$$\begin{bmatrix} u \\ \times \end{bmatrix} := W_{2n}(q \cdot t), \quad \begin{bmatrix} \times \\ v \end{bmatrix} := W_{2n}(p \cdot t),$$

$$z := F_{2n} \begin{bmatrix} u \\ 0 \end{bmatrix}, \quad w := F_{2n} \begin{bmatrix} 0 \\ v \end{bmatrix},$$

$$\begin{bmatrix} s - r \\ \times \end{bmatrix} := W_{2n}(q \cdot w - p \cdot z)$$

Thus, $x = M^{-1} b$ can be computed in $8\,\tau(2n) + O(n)$ computations. This is the implementation described in [8]. However, one of these FFTs can be eliminated using the observation that $p = \overline{q}$. This follows from the fact that

$$\begin{bmatrix} r_0 \\ e_0 \end{bmatrix} = K_{2n} \begin{bmatrix} 0 \\ r_1 \end{bmatrix}$$

and $F_n K_n = \overline{F}_n$, where $K_n = [e_0, e_{n-1}, \ldots, e_1]$ denotes the *reflection matrix* of order n. Thus, the implementation of the Gohberg-Semencul formula requires at most $7\,\tau(2n)+O(n) = 14\,\tau(n)+O(n)$ arithmetic operations.

Let us now write (2) as

$$\delta_{n-1} M^{-1} = A = T_1^T C_1 - T_0 C_0 ,$$

where T_1 and T_0 are as above, $C_0 = C(r_1)$ and $C_1 = C_0^T = C(K_n\, r_1)$. Define $u = C_0\, b$, $v = C_1\, b$, $r = T_0\, u$ and $s = T_1^T\, v$. Then the following convolution formulas can be used to calculate r and s.

$$u := r_1 * b \ , \quad v := (K_n\, r_1) * b \ ,$$

$$\begin{bmatrix} r \\ \times \end{bmatrix} := \begin{bmatrix} r_0 \\ e_0 \end{bmatrix} * \begin{bmatrix} u \\ 0 \end{bmatrix}, \quad \begin{bmatrix} s \\ \times \end{bmatrix} := \begin{bmatrix} 0 \\ r_1 \end{bmatrix} * \begin{bmatrix} v \\ 0 \end{bmatrix}.$$

Note that $F_n (K_n\, r_1) = \overline{F_n\, r_1}$, so in terms of FFTs, we have

$$t := F_n\, b \ , \quad p := F_n\, r_1 \, , \quad u := W_n (p \cdot t) \, , \quad v := W_n (\bar{p} \cdot t) \, ,$$

$$q := F_{2n} \begin{bmatrix} 0 \\ r_1 \end{bmatrix}, \quad z := F_{2n} \begin{bmatrix} u \\ 0 \end{bmatrix}, \quad w := F_{2n} \begin{bmatrix} 0 \\ v \end{bmatrix}, \tag{6}$$

$$\begin{bmatrix} s - r \\ \times \end{bmatrix} := W_{2n} (q \cdot w - \bar{q} \cdot z) \, .$$

These computations require $4\,\tau(n) + 4\,\tau(2n) + O(n) = 12\,\tau(n) + O(n)$. However, we can reduce this operation count further as follows.

Define the permutation matrix P_{2n} by $P_{2n}^T\, y = \begin{bmatrix} y_0' \\ y_1' \end{bmatrix}$, where $y_0' = [\eta_{2j}]_0^{n-1}$ and $y_1' = [\eta_{2j+1}]_0^{n-1}$ are the *even-* and *odd-indexed parts* of $y = [\eta_j]_0^{2n-1}$, respectively. Let $y = F_{2n} \begin{bmatrix} x_0 \\ x_1 \end{bmatrix}$, where x_0 and x_1 are n-vectors. Then it is easy to see that

$$\begin{bmatrix} y_0' \\ y_1' \end{bmatrix} = P_{2n}^T\, y = \begin{bmatrix} F_n & F_n \\ F_n D_n' & -F_n D_n' \end{bmatrix} \begin{bmatrix} x_0 \\ x_1 \end{bmatrix} = \begin{bmatrix} F_n (x_0 + x_1) \\ F_n\, D_n' (x_0 - x_1) \end{bmatrix}.$$

where $D_n' = \text{diag}[\omega_{2n}^{-k}]_{k=0}^{n-1}$. Thus,

$$P_{2n}^T\, F_{2n} \begin{bmatrix} x \\ 0 \end{bmatrix} = \begin{bmatrix} F_n\, x \\ F_n\, D_n'\, x \end{bmatrix},$$

$$P_{2n}^T\, F_{2n} \begin{bmatrix} 0 \\ x \end{bmatrix} = \begin{bmatrix} F_n\, x \\ -F_n\, D_n'\, x \end{bmatrix}.$$

It is shown in [2], and easily verified, that $F_n\, D_n'\, x$, where $x \in \mathbf{R}^n$, can be computed using one complex FFT of order $n/2$, which requires $\tau(n) + O(n)$ operations.

These formulas allow us to reduce the computation of the FFTs in display (6) as follows.

1. The jth component of p is the $2j$th component of q, so the so only q needs to be calculated, saving $\tau(n)$.

2. z can be obtained from $p \cdot t$ and u using $\tau(n) + O(n)$ operations, saving roughly $\tau(n)$ operations. The same observation holds for the computation of w from $\overline{p} \cdot t$ and v.

Consequently, the computation of $x = M^{-1} b$ using our modification of the Gohberg-Semencul formula requires at most $5\,\tau(n) + 2\,\tau(2n) + O(n)$ or $9\tau(n) + O(n)$ computations. This represents a computational savings of $5/14$ over the implementation of the Gohberg-Semencul formula as described above, neglecting the $O(n)$ terms.

Numerical experiments show that a savings of 5/14 (about 36%) in CPU time is indeed achieved. The results are summarized in Table 1, which shows average CPU times for the implementations of the Gohberg-Semencul formula (GS) and our circulant variant of the Gohberg-Semencul formula (CGS) as described above. Also displayed are the ratios of the average times used by CGS to those of GS. The experiments were performed on the VAX 11/750 at Northern Illinois University.

Timing comparison (CPU seconds) Gohberg-Semencul (GS) vs. Circulant variant (CGS)			
n	CGS	GS	CGS/GS
4	0.007167	0.010833	0.662
8	0.015000	0.023667	0.634
16	0.0318334	0.047667	0.668
32	0.0645001	0.095167	0.678
64	0.130334	0.201834	0.646
128	0.264834	0.414335	0.639
256	0.558001	0.877004	0.636
512	1.18350	1.85517	0.638
1024	2.49901	3.90801	0.639
2048	5.28350	8.25551	0.640
4096	11.1670	17.3942	0.642
8192	23.6947	36.8251	0.643
16384	50.8578	79.1816	0.642

References

[1] G. S. Ammar and W. B. Gragg, *The Generalized Schur Algorithm for the Superfast Solution of Toeplitz Systems,* in Rational Approximation and its Applications in Mathematics and Physics (J. Gilewicz, M. Pindor and W. Siemaszko, editors), Lecture Notes in Mathematics 1237, Springer-Verlag, Berlin, 1987, 315-330.

[2] G.S. Ammar and W.B. Gragg, *Superfast Solution of Real Positive Definite Toeplitz Systems,* SIAM J. Matrix Anal. Appl., 9 (1988), 61-76.

[3] B. Friedlander, M. Morf, T. Kailath and L. Ljung, *New Inversion Formulas for Matrices Classified in Terms of Their Distance from Toeplitz Matrices,* Lin. Alg. and Appl., 27 (1979), 31-60.

[4] P. Gader, *Displacement Operator Based Decompositions of Matrices Using Circulants or Other Group Matrices,* Lin. Alg. and Appl., submitted for publication.

[5] I. C. Gohberg and I. A. Fel′dman, Convolution Equations and Projection Methods for their Solution, American Mathematical Society, Providence, RI, 1974.

[6] G. H. Golub and C. F. Van Loan, Matrix Computations, Johns Hopkins University Press, Baltimore, MD, 1984.

[7] I.S. Iohvidov, Hankel and Toeplitz Matrices and Forms, Algebraic Theory, Birkhäuser, Boston, MA, 1982.

[8] J. R. Jain, *An Efficient Algorithm for a Large Toeplitz Set of Linear Equations,* IEEE Trans. Acoust. Speech Signal Process., 27 (1979), 612-615.

[9] T. Kailath, S.Y. Kung and M. Morf, *Displacement Ranks of Matrices and Linear Equations,* J. Math. Anal. Appl., 68 (1979), 395-407.

[10] T. Kailath, A. Vieira and M. Morf, *Inverses of Toeplitz Operators, Innovations, and Orthogonal Polynomials,* SIAM Rev., 20 (1978), 106-119.

[11] W. Trench, *An Algorithm for the Inversion of Finite Toeplitz Matrices,* SIAM J. Appl. Math., 12 (1964), 515-522.