MATHEMATICS OF COMPUTATION
VOLUME 55, NUMBER 191
JULY 1990, PAGES 179-190

ON COMPUTATIONS WITH DENSE STRUCTURED MATRICES

VICTOR PAN

ABSTRACT. We reduce several computations with Hilbert and Vandermonde type matrices to matrix computations of the Hankel-Toeplitz type (and vice versa). This unifies various known algorithms for computations with dense structured matrices and enables us to extend any progress in computations with matrices of one class to the computations with other classes of matrices. In particular, this enables us to compute the inverses and the determinants of $n \times n$ matrices of Vandermonde and Hilbert types for the cost of $O(n \log^2 n)$ arithmetic operations. (Previously, such results were only known for the more narrow class of Vandermonde and generalized Hilbert matrices.)

1. INTRODUCTION

The important concepts of the *displacement ranks* and the *displacement generators* of matrices having Toeplitz-like or Hankel-like structure were introduced about a decade ago (see [17, 18]) and turned out to be effective tools for the inversion and factorization of such matrices (see [6, 20, 2, 9, 16, 25]). In [15, 14, 13, 28], similar tools (including *scaling generators*) were used for computations with Hilbert-like and Vandermonde-like matrices, having important applications, in particular, to integral equations and conformal mappings (see [13, 14, 21]). Since the tools for devising algorithms for computations for the three classes of matrices (of Toeplitz-Hankel, Hilbert and Vandermonde types) have many common features, one may hope to reduce computations for each such class of matrices to computations for any other of these classes. In particular, we may hope to improve Hilbert and Vandermonde type computations by reducing them to Toeplitz-Hankel type computations, where more effective algorithms are available. [8] contains a successful ad hoc result of such kind for Vandermonde linear systems.

In our present paper, we systematically reduce the algorithms for all the three classes of matrices to each other by relying on a general tool given by an extended version of Lemma 3 from [10]. To demonstrate the power of our approach, we substantially improve the known methods for computing the inverses and the determinants of $n \times n$ matrices of Vandermonde and Hilbert types, yielding the

Received January 17, 1989.
1980 *Mathematics Subject Classification* (1985 *Revision*). Primary 47B35, 65F05, 15A09.
Key words and phrases. Dense structured matrices, algorithms, displacement rank, Hilbert, Vandermonde, Toeplitz and Hankel matrices.
The author was supported by NSF Grant CCR-8805782 and by PSC CUNY Award #668541.

cost bounds of $O(n\log^2 n)$ arithmetic operations. So far, such bounds have only been obtained for Vandermonde and generalized Hilbert matrices [11, 8] which constitute a more narrow class. The approach promises further applications, which we briefly discuss in §7.

Our computations involve evaluation of a polynomial at several points and interpolation by a polynomial; to avoid the numerical stability problems at these stages, one may apply the recent fast approximation algorithm of [27]; it is *not* clear if such problems can be avoided at all stages of our algorithms, unless the algorithms are implemented using computer algebra subroutines.

We will organize our paper as follows: we will recall some known facts and concepts for computations with structured matrices (together with some simple extensions of these facts and concepts) in §§2–5, we will adapt and extend the above-mentioned result from [10] in §6, we will discuss its applications in §7, and we will show how to improve computing the inverses and the determinants of matrices of Vandermonde and Hilbert types in §§8–11.

2. Definitions

Hereafter, $\mathbf{R}_{p,q}$ denotes the space of $p\times q$ matrices, which are vectors if $p=1$ or $q=1$. W^{T} denotes the transpose of a matrix or of a vector W. $W^{-\mathrm{T}}$ denotes the inverse of a matrix W^{T}. $\mathbf{0}$ denotes the null vector, $\mathbf{e}$ denotes the vector $[1,1,\dots,1]^{\mathrm{T}}$, and $\mathbf{u}(k)$ denotes the kth unit coordinate vector of appropriate dimension. For a vector $\mathbf{a}=[a_0,\dots,a_{n-1}]^{\mathrm{T}}$ of dimension n, $D(\mathbf{a})=\operatorname{diag}(a_0,\dots,a_{n-1})$ denotes the $n\times n$ diagonal matrix with the diagonal entries $a_0,\dots,a_{n-1}$. $O=D(\mathbf{0})$ and $I=D(\mathbf{e})$ denote the null and the identity matrices, respectively. Z denotes the square matrix, zero everywhere, except for its first subdiagonal filled with ones, so that premultiplications and postmultiplications by Z and by Z^{T} shift the entries of a matrix $A=[a_{ij}]$ as follows:

$$ZA=[a_{i-1,j}] \quad \text{(shift down)}, \tag{2.1}$$

$$AZ=[a_{i,j-1}] \quad \text{(shift left)}, \tag{2.2}$$

$$Z^{\mathrm{T}}A=[a_{i+1,j}] \quad \text{(shift up)}, \tag{2.3}$$

$$AZ^{\mathrm{T}}=[a_{i,j+1}] \quad \text{(shift right)}, \tag{2.4}$$

where $a_{pq}=0$ for p and/or q out of range.

Next, we will recall some fundamental concepts of computations with structured matrices (compare the examples in the next section).

Definition 2.1. Let $F(A)$ denote the image of an operator F applied to an $m\times n$ matrix A, and let $G\in\mathbf{R}_{m,d}$ and $H\in\mathbf{R}_{n,d}$ denote two matrices such that $F(A)=GH^{\mathrm{T}}$. Then the rank $r=r(F(A))$ of the matrix $F(A)$ is called the *F-rank of A*, and the pair of matrices G and H is called an *F-generator of*

A of length d. (The rank of A is the I-rank of A and a generator of A is an I-generator of A for the identity operator I such that $IA = A$ for all A.)

For a fixed operator F and a fixed matrix A, an F-generator of A of minimum length r can be computed and can also be approximated by lower-rank matrices by using the singular value decomposition of $F(A)$ (see [23]).

3. Displacement operators of Hankel-Toeplitz type

Following [17, 18], consider the operators:

$$F = F(Z, Z^{\mathrm{T}}), \qquad F(A) = A - ZAZ^{\mathrm{T}}, \tag{3.1}$$

$$F = F(Z^{\mathrm{T}}, Z), \qquad F(A) = A - Z^{\mathrm{T}}AZ, \tag{3.2}$$

$$F = F(Z, Z), \qquad F(A) = A - ZAZ, \tag{3.3}$$

$$F = F(Z^{\mathrm{T}}, Z^{\mathrm{T}}), \qquad F(A) = A - Z^{\mathrm{T}}AZ^{\mathrm{T}}. \tag{3.4}$$

Hereafter, we will use the notation $F(Z, Z)(A)$ to designate $F(A)$ for $F = F(Z, Z)$.

Because of the shifting properties (2.1)–(2.4) of multiplications by Z and Z^{T}, the operators F of (3.1) and (3.2) zero all the entries of a Toeplitz matrix A, except for its first row and column under (3.1) and for its last row and column under (3.2), which are invariant in the transition from A to $F(A)$. This defines F-generators of A of length at most 2 in such cases and, similarly, in the cases where A is a Hankel matrix and F is an operator of (3.3) or (3.4). We have the following simple relations:

Fact 3.1. *There holds $F(A^{\mathrm{T}}) = F^{\mathrm{T}}(A)$ for the operators F of (3.1) and (3.2); similarly, $F(Z, Z)(A^{\mathrm{T}}) = F^{\mathrm{T}}(Z^{\mathrm{T}}, Z^{\mathrm{T}})(A)$.*

The two next (rather simple but fundamental) theorems are due to, or implicit in, [17].

Theorem 3.1. *For any nonsingular matrix A,*

$$\operatorname{rank} F(Z, Z^{\mathrm{T}})(A) = \operatorname{rank} F(Z^{\mathrm{T}}, Z)(A^{-1}),$$
$$\operatorname{rank} F(Z, Z)(A) = \operatorname{rank} F(Z, Z)(A^{-1}),$$
$$\operatorname{rank} F(Z^{\mathrm{T}}, Z^{\mathrm{T}})(A) = \operatorname{rank} F(Z^{\mathrm{T}}, Z^{\mathrm{T}})(A^{-1}).$$

Theorem 3.2. *For the operators F of (3.1)–(3.4),*

$$F(A) = GH^{\mathrm{T}} = \sum_{i=1}^{d} \mathbf{g}_i \mathbf{h}_i^{\mathrm{T}}, \qquad G = [\mathbf{g}_1, \dots, \mathbf{g}_d], \; H = [\mathbf{h}_1, \dots, \mathbf{h}_d],$$

if and only if

$$A = \sum_{i=1}^{d} L(\mathbf{g}_i)L^{\mathrm{T}}(\mathbf{h}_i), \quad \textit{under } (3.1),$$

$$A = \sum_{i=1}^{d} L^{\mathrm{T}}(\mathbf{g}_i^*)L(\mathbf{h}_i^*), \quad \textit{under } (3.2),$$

$$A = \sum_{i=1}^{d} L(\mathbf{g}_i)L^{\mathrm{T}^*}(\mathbf{h}_i), \quad \textit{under } (3.3),$$

$$A = \sum_{i=1}^{d} L^*(\mathbf{g}_i)L^{\mathrm{T}}(\mathbf{h}_i), \quad \textit{under } (3.4).$$

Here (and hereafter), $L(\mathbf{x})$ denotes the lower triangular Toeplitz matrix with the first column $\mathbf{x}$; $L^{\mathrm{T}^*}(\mathbf{x})$ and $L^*(\mathbf{y})$ denote the upper and lower triangular Hankel matrices such that the first row of $L^{\mathrm{T}^*}(\mathbf{x})$ is $\mathbf{x}^{\mathrm{T}}$, the last row of $L^*(\mathbf{y})$ is $\mathbf{y}^{\mathrm{T}}$, the entries of $L^{\mathrm{T}^*}(\mathbf{x})$ below the antidiagonal and the entries of $L^*(\mathbf{y})$ above the antidiagonal are zeros, and the vectors $\mathbf{g}_i^*$ and $\mathbf{h}_i^*$ are just the vectors $\mathbf{g}_i$ and $\mathbf{h}_i$ with the entries in reverse order.

4. Scaling operators of Hilbert type

We will follow [13–15] and, for a pair of n-dimensional vectors $\mathbf{s}$ and $\mathbf{t}$, will define the operators

(4.1) $$F = F_{\mathbf{s},\mathbf{t}}, \qquad F(A) = D(\mathbf{s})A - AD(\mathbf{t}).$$

For motivation, let us apply such an operator to the matrix

(4.2) $$A = [a_{ij}], \quad \text{where } 1/a_{ij} = s_i - t_j \text{ for all } i \text{ and } j.$$

(This matrix generalizes the Hilbert matrix $A = [1/(i+j+1)]$.) Then

(4.3) $$F_{\mathbf{s},\mathbf{t}}(A) = \mathbf{e}\mathbf{e}^{\mathrm{T}}.$$

Definition 4.1. An operator $\widehat{F}$ is called *dual* to an operator F if $\widehat{F}(A^{\mathrm{T}}) = -F(A)^{\mathrm{T}}$ and if, for every nonsingular matrix A, $\widehat{F}(A^{-1}) = -A^{-1}F(A)A^{-1}$.

Surely, for the operator $\widehat{F}$ dual to an operator F, $\operatorname{rank} F(A) = \operatorname{rank} \widehat{F}(A^{\mathrm{T}})$; furthermore, $\operatorname{rank} \widehat{F}(A^{-1}) = \operatorname{rank} F(A)$ provided that A is a nonsingular matrix and, if F is also dual to $\widehat{F}$, then $F(A^{-\mathrm{T}}) = A^{-\mathrm{T}}HG^{\mathrm{T}}A^{-\mathrm{T}}$, $\operatorname{rank} F(A^{-\mathrm{T}}) = \operatorname{rank} F(A)$.

The next two simple facts extend Theorems 3.1 and 3.2 to the case of the operators $F_{\mathbf{s},\mathbf{t}}$:

Fact 4.1. *The operators $F_{\mathbf{s},\mathbf{t}}$ of (4.1) and $F_{\mathbf{t},\mathbf{s}}$ are dual to each other for any fixed pair of vectors $\mathbf{s}$ and $\mathbf{t}$ of the same dimension.*

Fact 4.2. *Let $F = F_{\mathbf{s},\mathbf{t}}$, $A = [a_{ij}]$, and $F(A) = [f_{ij}]$. Then $f_{ij} = (s_i - t_j)a_{ij}$ for all i and j.*

Combining equation (4.3) and Facts 4.1 and 4.2 implies the following results:

Corollary 4.1. *Let A denote the $n \times n$ nonsingular generalized Hilbert matrix of (4.2) for some fixed vectors $\mathbf{s}$ and $\mathbf{t}$, $B = [b_{ij}] = A^{-1}$, $F = [f_{ij}] = F_{\mathbf{t},\mathbf{s}}(B)$. Then all the components of the vector $B\mathbf{e}$ are nonzero, the matrix F has rank 1, $F = -B\mathbf{e}\mathbf{e}^{\mathrm{T}}B$, and $b_{ij} = f_{ij}/(t_i - s_j)$ for all i and j; furthermore, every fixed pair of a row and a column of the matrix B can be recovered from its first row and column by using $O(n)$ arithmetic operations.*

Finally, assume that the vectors $\mathbf{s}$ and/or $\mathbf{t}$ have no zero components and apply *scaling* to reduce the operator $F_{\mathbf{s},\mathbf{t}}$ to the format

$$F = F(K, L), \qquad F(A) = A - KAL \tag{4.4}$$

for appropriate matrices K and L. Specifically, consider the operators $F(D^{-1}(\mathbf{s}), D(\mathbf{t}))$ and $F(D(\mathbf{s}), D^{-1}(\mathbf{t}))$, which have the format (4.4) and are defined by scaling the equation (4.1) and the respective matrices of the F-generators of A by the factors $D^{-1}(\mathbf{s})$ and $D^{-1}(\mathbf{t})$. Such scaling does not change the F-rank of A.

Note that (4.4) is also a format for the operators F of (3.1)–(3.4).

5. Operators of displacement and scaling of Vandermonde type

For a vector $\mathbf{v} = [v_j]$, consider the four following operators (compare [13–15]):

$$F = F_{\mathbf{v},Z}, \qquad F(A) = D(\mathbf{v})A - AZ, \tag{5.1}$$

$$F = F_{\mathbf{v},Z^{\mathrm{T}}}, \qquad F(A) = D(\mathbf{v})A - AZ^{\mathrm{T}}, \tag{5.2}$$

$$F = F_{Z,\mathbf{v}}, \qquad F(A) = ZA - AD(\mathbf{v}), \tag{5.3}$$

$$F = F_{Z^{\mathrm{T}},\mathbf{v}}, \qquad F(A) = Z^{\mathrm{T}}A - AD(\mathbf{v}). \tag{5.4}$$

Let A be a Vandermonde matrix:

$$A = [a_{ij}], \qquad a_{ij} = v_i^j \text{ for } i, j = 0, 1, \ldots, n-1. \tag{5.5}$$

Then we have

$$F_{\mathbf{v},Z}(A) = D^{n-1}(\mathbf{v})\mathbf{e}\mathbf{u}^{\mathrm{T}}(n-1); \tag{5.6}$$

if, in addition, $\mathbf{v}_i \neq 0$ for all i, and if $\mathbf{v}^{-1}$ denotes the vector $[1/v_i]$, then

$$F_{\mathbf{v}^{-1},Z^{\mathrm{T}}}(A) = D^{-1}(\mathbf{v})\mathbf{e}\mathbf{u}^{\mathrm{T}}(0). \tag{5.7}$$

Fact 4.1 is immediately extended as follows:

Fact 5.1. *The operators of the two following pairs are dual to each other (for any fixed vector $\mathbf{v}$): $F_{\mathbf{v},Z}$ and $F_{Z,\mathbf{v}}$, $F_{\mathbf{v},Z^{\mathrm{T}}}$ and $F_{Z^{\mathrm{T}},\mathbf{v}}$.*

Next, we will extend Fact 4.2.

Fact 5.2. *Let* $A = [a_{ij}]$, $F(A) = [f_{ij}]$, *and* $\mathbf{v} = [v_i]$. *Then*

$$\begin{aligned} f_{ij} &= v_i a_{ij} - a_{i,j+1} & \text{for } F = F_{\mathbf{v},Z}, \\ f_{ij} &= v_i a_{ij} - a_{i,j-1} & \text{for } F = F_{\mathbf{v},Z^{\mathrm{T}}}, \\ f_{ij} &= a_{i-1,j} - v_j a_{ij} & \text{for } F = F_{Z,\mathbf{v}}, \\ f_{ij} &= a_{i+1,j} - v_j a_{ij} & \text{for } F = F_{Z^{\mathrm{T}},\mathbf{v}}, \end{aligned}$$

where $a_{pq} = 0$ *if* p *and/or* q *are out of range.*

Fact 5.3 [14]. *Let* $F = F_{Z,\mathbf{v}}$ *denote the operator of* (5.3), $F_{Z,\mathbf{v}}(A) = -GH^{\mathrm{T}}$, $G = [\mathbf{g}_1, \dots, \mathbf{g}_r]$, $H = [\mathbf{h}_1, \dots, \mathbf{h}_r]$, *and let* $V(\mathbf{v}^{-1}) = [v_i^{-j}]$. *Then*

$$A = \sum_{i=1}^{r} L(\mathbf{g}_i) D^{-1}(\mathbf{v}) V^{\mathrm{T}}(\mathbf{v}^{-1}) D(\mathbf{h}_i).$$

Expressions similar to the last one (defining the reversion of the operator F of (5.3)) also hold for the reversion of the operators F of (5.1), (5.2), and (5.4); this extension immediately follows from Fact 5.1 and enables us to recover (Vandermonde-like) matrices A from their images $F(A)$ for such operators F. We will omit these explicit formulae, as well as the extension of Corollary 4.1.

Finally, the operators of (5.1)–(5.4) can be immediately reduced to the form (4.4) by means of their scaling by the matrix $D^{-1}(\mathbf{v})$.

6. Computing a generator of a matrix product

Since we know how to recover a matrix from its F-generators for the operators F of the previous sections, we may save such F-generators rather than the matrix itself; this is economical if the lengths of the F-generators are small.

Suppose that we need to perform some arithmetic operations with the matrices stored this way. Do we have to recover the entries of the input matrices? Not necessarily. In many cases, we will be better off if we operate with the generators to the very end of the computations; we will assume this hereafter. The additions and subtractions of matrices will be reduced to the union operations with their F-generators. For the multiplication of matrices, we will use the following extension of a result from [10]:

Proposition 6.1. *For seven matrices* A, B, K, L, M, N *and* Δ, *and for three operators* F, F_1 *and* F_2 *such that*

$$F_1(A) = A - KAL, \qquad F_2(B) = B - MBN, \tag{6.1}$$

$$F(AB) = AB - KABN, \qquad \Delta = LM - I, \tag{6.2}$$

the following matrix equations hold:

$$F(AB) = 0.5F_1(A)(2B - F_2(B)) + 0.5(2A - F_1(A))F_2(B) + \Delta_0, \tag{6.3}$$

$$\Delta_0 = KA\Delta BN. \tag{6.4}$$

Proof. Since $F(AB) = AB - KAIBN$ and $I = LM - \Delta$, we have $F(AB) = AB - KALMBN + \Delta_0$ (see (6.2) and (6.4)). Therefore,

$$F(AB) = 0.5(A - KAL)(B + MBN) + 0.5(A + KAL)(B - MBN) + \Delta_0.$$

Substitute the equations (6.1), observe that $B + MBN = 2B - F_2(B)$ and $A + KAL = 2A - F_1(A)$, and deduce (6.3) to prove the proposition. □

Given the matrices A and B and the operators F_1 and F_2 of (6.1), we may compute the image $F(AB)$ of the operator F of (6.2) by applying equations (6.3) and (6.4). We will refer to this computation as *Algorithm* 6.1.

To estimate its cost, let us set $r = \operatorname{rank} F(AB)$, $r_0 = \operatorname{rank}\Delta$, $r_1 = \operatorname{rank} F_1(A)$, and $r_2 = \operatorname{rank} F_2(B)$. Then it follows that $r \le r_0 + r_1 + r_2$.

For smaller r_0, r_1, and r_2, Algorithm 6.1 is essentially reduced to computing the I-generators of the matrix products $F_1(A)B$ and $AF_2(B)$, which amounts to r_1 multiplications of vectors by the matrix B and r_2 multiplications of the matrix A by vectors.

To apply Proposition 6.1 and Algorithm 6.1, we need to represent the operators F_1 and F_2 in the form (4.4) (we will use scaling in the cases (4.1), (5.1)–(5.4)) and to choose matrices L and M for F_1 and F_2, respectively, so as to keep the rank of the matrix $\Delta = LM - I$ smaller. Here are three relevant choices:

$$L = Z, \quad M = Z^{\mathrm{T}}, \quad \Delta = -\mathbf{u}(0)\mathbf{u}^{\mathrm{T}}(0), \quad r_0 = 1, \tag{6.5}$$

$$L = Z^{\mathrm{T}}, \quad M = Z, \quad \Delta = -\mathbf{u}(n-1)\mathbf{u}^{\mathrm{T}}(n-1), \quad r_0 = 1, \tag{6.6}$$

$$L = D(\mathbf{v}), \quad M = D^{-1}(\mathbf{v}), \quad \Delta = 0, \quad r_0 = 0. \tag{6.7}$$

7. Applications of Proposition 6.1 (outline)

Proposition 6.1 is implicit in [10] in the proof of an estimate for the F-rank of the product AB, given F-ranks of the factors A and B, where F is the operator (3.1) applied to A, B, and AB. We regard this proposition as a more general tool. In particular, it enables us to reduce the computations for one class of matrices to another. For example, let HT stand for a matrix of Hankel or Toeplitz type, H for a matrix of Hilbert type and V for a matrix of Vandermonde type. These definitions just mean that such a matrix has small F-rank, where the operator F is defined by (3.1)–(3.4), (4.1), or (5.1)–(5.4), respectively.

Then Proposition 6.1 and the equations (6.5)–(6.7) enable us to make transition between the classes of matrices HT, H, and V according to the following Table 7.1.

TABLE 7.1

A	B	AB	F-rank r of AB
HT	V	V	$r \le r_1 + r_2 + 1$
V	V	H	$r \le r_1 + r_2 + 1$
V	V	HT	$r \le r_1 + r_2$
H	V	V	$r \le r_1 + r_2$
HT	HT	HT	$r \le r_1 + r_2 + 1$
H	H	H	$r \le r_1 + r_2$

This transition requires, of course, that the operators F_1 and F_2 be reduced to the format (6.1) with the matrices L and M reconciled with each other so as to ensure (6.5), (6.6), or (6.7).

Now, let us be given, say, an $F_{\mathbf{s},\mathbf{t}}$-generator of length r_1 for a matrix A of Hilbert type (of F-rank r_1) and for the operator $F_{\mathbf{s},\mathbf{t}}$ of (4.1), and let us seek A^{-1} and/or $\det A$. Then we may choose the Vandermonde matrix $B = [1/t_i^j]$ such that AB is a matrix of F-rank at most $r_1 + 1$ for F of (5.1) with $\mathbf{v} = \mathbf{s}^{-1}$, so that $\mathbf{v} = [v_i] = \mathbf{s}^{-1} = [1/s_i]$, where $\mathbf{s} = [s_i]$; then compute an F-generator of AB of length at most $r_1 + 1$ by using Algorithm 6.1; then invert the matrix AB and/or compute $\det(AB)$ and $\det B$, and, finally, compute an $\widehat{F}_{\mathbf{t},\mathbf{s}}$-generator of $A^{-1} = B(AB)^{-1}$ of length $\mathbf{r}_1$ and/or $\det A = \det(AB)/\det B$. Since B is a Vandermonde matrix and AB is a matrix of Vandermonde type, we reduce Hilbert type computations to the computations of Vandermonde type. Similar transition is possible from any of the classes HT, H, and V to any other such class (see Table 7.1). The last two lines of Table 7.1 also indicate that we may make transitions within the classes HT and H, so that the associated operators F change as we like.

In the next sections, we will specify a reduction of the classes H and V to HT, which will imply an improvement of the known algorithms for the classes H and V. Most important in applied mathematics and in computational practice are the more narrow subclasses of Vandermonde and generalized Hilbert matrices (see the introduction), but for these subclasses good competitive algorithms are available [11, 8]. We will still restrict our demonstration to the latter narrow subclasses because the extension of our algorithms from these subclasses to the more general classes of Hilbert type and Vandermonde type matrices is immediate, due to Facts 4.2, 5.2, and 5.3.

Remark 7.1. Other pairs of matrices L and M, in addition to the ones of (6.5)–(6.7), may suggest further interesting applications of Proposition 6.1. Also, its plausible extensions (for instance, to the operators F_1 and F_2 having the commutant form $KA - AL$ with singular matrices K and L) would be useful.

Remark 7.2. *Hankel-Toeplitz computations.* The known effective algorithms of [20, 2, 9, 23, 24, 25] invert and recursively factorize Toeplitz and Hankel matrices, as well as all the matrices given with their F-generators of small length for the operators F of (3.1)–(3.4), and this gives the determinants of such matrices too. The overall cost of these computations is $O(n\log^2 n)$ arithmetic operations, provided that all the leading principal submatrices of the input matrix are nonsingular.

8. Computing the determinant of a Vandermonde matrix

Let A be a Vandermonde matrix of (5.5). Then $A^{\mathrm{T}}A$ is the Hankel matrix:

$$A^{\mathrm{T}}A = [h_{ij}], \qquad h_{ij} = \sum_{k=0}^{n-1} v_k^{i+j} \text{ for all } i \text{ and } j. \tag{8.1}$$

To compute $A^{\mathrm{T}}A$ given A, we will first compute the coefficients of the polynomial

$$p_n(x) = \prod_{k=0}^{n-1}(x - v_k) \tag{8.2}$$

(see [1, 7]) and then obtain all the entries h_{ij} of $A^{\mathrm{T}}A$ by solving the triangular Toeplitz system of Newton's identities. We will then compute $\det(A^{\mathrm{T}}A) = (\det A)^2$ and will recover $\det A$ (within the factor -1). Using the known algorithms for Hankel-Toeplitz computations, we will compute $\det A$ for the overall cost of $O(n\log^2 n)$ arithmetic operations. This algorithm will be referred to as *Algorithm* 8.1. It is actually due to [8], although only the cost bound $O(n^2\log^2 n)$ has been deduced there.

9. Computing the determinant of a generalized Hilbert matrix

Let $\mathrm{H} = A$ denote a generalized Hilbert matrix of (4.2), and let us first compute an F-generator of the matrix

$$S = \mathrm{V}^{\mathrm{T}}(\mathbf{t}^{-1})\,\mathrm{H}\mathrm{V}(\mathbf{s}), \tag{9.1}$$

where $\mathrm{V}(\mathbf{w})$, for a vector $\mathbf{w}$, denotes the Vandermonde matrix $[w_i^j]$, $\mathbf{t}^{-1}$ denotes the vector $[1/t_i]$, and F denotes a fixed operator of (3.1)–(3.4), say, $F(Z, Z^{\mathrm{T}})$, to be certain. Without loss of generality, we choose the vectors $\mathbf{s}$ and $\mathbf{t}$ (which define $\mathrm{H} = A$ via (4.2)) to have no zero coordinates. For appropriate operators F, we will twice apply Algorithm 6.1 to compute:

(1) first an F-generator of the matrix $\mathrm{V}^{\mathrm{T}}(\mathbf{t}^{-1})\,\mathrm{H}$ of length 2 or less and

(2) then an F-generator of S of length 3 or less.

Let us specify the input to Algorithm 6.1 in these applications:

(1) $K = Z$, $L = D^{-1}(\mathbf{t})$, $M = L^{-1}$, $N = D(\mathbf{s})$, $A = \mathrm{V}^{\mathrm{T}}(\mathbf{t}^{-1})$, $B = H$,

(2) $K = Z$, $L = D(\mathbf{s})$, $M = L^{-1}$, $N = Z^{\mathrm{T}}$, $A = \mathrm{V}^{\mathrm{T}}(\mathbf{t}^{-1})\,\mathrm{H}$, $B = \mathrm{V}(\mathbf{s})$.

Our estimates for the F-ranks of Vandermonde and generalized Hilbert matrices for the operators F of (4.1), (5.1)–(5.4) (valid also after scaling the operator F) and Proposition 6.1 imply the following bounds at stages(1) and (2):

(1) $\Delta = O$, $r_0 = 0$, $r_1 = r_2 = 1$, $r \le 2$,

(2) $\Delta = O$, $r_0 = 0$, $r_1 \le 2$, $r_2 = 1$, $r \le 3$.

In the second application of Algorithm 6.1, an F-generator of length 3 (or less) for S is output, where F is the operator of (3.1), so that, applying the known algorithms for Toeplitz-Hankel computations, we will immediately compute $\det S = \det \mathrm{V}^{\mathrm{T}}(\mathbf{t}^{-1}) \det \mathrm{H} \det \mathrm{V}(\mathbf{s})$.

Then we will compute the determinants of the two Vandermonde matrices $\mathrm{V}(\mathbf{t})$ and $\mathrm{V}(\mathbf{s})$ by applying Algorithm 8.1, and this will give us

$$\det \mathrm{H} = \det S \det \mathrm{V}(\mathbf{t}) / \det \mathrm{V}(\mathbf{s})$$

for the overall cost of the two applications of each of Algorithms 6.1 and 8.1. We refer to §8 for Algorithm 8.1 and estimate the cost of the applications of Algorithm 6.1. In view of Proposition 6.1, this cost is dominated by the cost of premultiplying each of the matrices $\mathrm{V}(\mathbf{s})$ (once) and $A\,\mathrm{V}(\mathbf{s})$ (once) and of postmultiplying the matrix A (once) and the matrix $\mathrm{V}^{\mathrm{T}}(\mathbf{t}^{-1})$ (twice) by vectors. This amounts to:

(a) two multiplications of vectors by the generalized Hilbert matrices A and A^{T} and
(b) four multiplications of vectors by the Vandermonde matrices $\mathrm{V}(\mathbf{s})$ and $\mathrm{V}(\mathbf{t}^{-1})$.

The computations of part (a) slightly generalize the well-known *Trummer's problem*, having several important applications. There are a few effective solution algorithms for part (a) (see [3, 4, 5, 12, 19, 22, 26]). In particular, the algorithm of [12] involves $O(n \log^2 n)$ arithmetic operations.

In part (b), we need to multiply a vector $\mathbf{w}^{\mathrm{T}}$ by a Vandermonde matrix A of (5.5). For this purpose, we will first compute the Hankel matrix $A^{\mathrm{T}}A$ of (8.1) (as in §8) and the vector $\mathbf{g}^{\mathrm{T}} = \mathbf{w}^{\mathrm{T}} A^{-\mathrm{T}}$ (the latter step is interpolation by a polynomial of degree $n-1$ or less). Then it will remain to multiply the vector $\mathbf{g}^{\mathrm{T}}$ (for the cost of 3 FFT's) by the Hankel matrix $A^{\mathrm{T}}A$ in order to arrive at the desired vector $\mathbf{w}^{\mathrm{T}}A$.

The overall cost of this algorithm is $O(n \log^2 n)$ arithmetic operations.

Remark 9.1. Here is an alternative way for the evaluation of the vector $\mathbf{w}^{\mathrm{T}}A$:

(1) Compute the Toeplitz matrix $B^{\mathrm{T}}A$, where $B = [b_{ij}]$, $b_{ij} = 1/v_i^j$, $A = [a_{ij}]$, $a_{ij} = v_i^j$, and $B^{\mathrm{T}}A = [t_{ij}]$, $t_{ij} = \sum_{k=0}^{n-1} v_k^{j-i}$. (First compute the coefficients of $p_n(x)$ of (8.2) and then solve two triangular Toeplitz systems of Newton's identities, each of half the size of the system of §8.)

(2) Compute the vector $\mathbf{h} = \mathbf{w}^{\mathrm{T}} B^{-\mathrm{T}} = (B^{-1}\mathbf{w})^{\mathrm{T}}$ by means of interpolation, for B is a Vandermonde matrix.

(3) Compute $\mathbf{w}^{\mathrm{T}}A$ by multiplying the vector $\mathbf{h}$ by the Toeplitz matrix B^{T}.

10. Inversion of a Vandermonde matrix

The algorithms of the previous section can be adapted in order to compute the solution vector $\mathbf{w}^{\mathrm{T}}A^{-1}$ of the transposed Vandermonde system of equations

$\mathbf{x}^{\mathrm{T}}A = \mathbf{w}^{\mathrm{T}}$. Indeed, first compute the Hankel matrix $A^{\mathrm{T}}A$, then solve the Hankel system $\mathbf{y}^{\mathrm{T}}A^{\mathrm{T}}A = \mathbf{w}^{\mathrm{T}}$ (using the known effective algorithms) and, finally, compute $\mathbf{w}^{\mathrm{T}}A^{-1} = \mathbf{y}^{\mathrm{T}}A^{\mathrm{T}} = (A\mathbf{y})^{\mathrm{T}}$. The latter step amounts to the evaluation of a degree $n-1$ polynomial at n points. Similarly, we may adapt the algorithm of Remark 9.1.

In view of Fact 5.1, to complete the evaluation of an F-generator of length 1 defined by (5.3) or (5.4) for the inverse of a Vandermonde matrix A, it remains to solve the linear system $A\mathbf{z} = \mathbf{w}$, which amounts to interpolation by a polynomial of degree $n-1$ or less. This yields the desired F-generator of A^{-1} of length 1 in $O(n\log^2 n)$ arithmetic operations.

11. Inversion of a generalized Hilbert matrix

To compute an F-generator of length 1 for the inverse of a nonsingular generalized Hilbert matrix A of (4.2), we will first compute an F-generator of length at most 3 for the matrix S of (9.1). Then we will compute an $\widehat{F}$-generator of length at most 3 for S^{-1} by means of the known efficient algorithms. We have the equation $A^{-1} = \mathrm{V}(\mathbf{s})S^{-1}\mathrm{V}^{\mathrm{T}}(\mathbf{t}^{-1})$, so that Corollary 4.1 implies that it suffices to compute the vectors $A^{-1}\mathbf{u}(0)$ and $\mathbf{u}^{\mathrm{T}}(0)A^{-1}$. Because of Theorem 3.2, six multiplications of triangular Toeplitz matrices by vectors suffice in order to compute each of the vectors $\mathbf{c} = S^{-1}\mathrm{V}^{\mathrm{T}}(\mathbf{t}^{-1})\mathbf{u}(0)$ and $\mathbf{d}^{\mathrm{T}} = \mathbf{u}^{\mathrm{T}}(0)\mathrm{V}(\mathbf{s})S^{-1}$. Then it will remain to compute the vectors $\mathrm{V}(\mathbf{s})\mathbf{c}$ and $\mathbf{d}^{\mathrm{T}}\mathrm{V}^{\mathrm{T}}(\mathbf{t}^{-1}) = (\mathrm{V}(\mathbf{t}^{-1})\mathbf{d})^{\mathrm{T}}$, which amounts to the evaluation at n points of each of two polynomials of degrees at most $n-1$. The overall asymptotic cost of computing A^{-1} in this case is again $O(n\log^2 n)$ arithmetic operations.

Acknowledgments

I am happy to thank the referee for helpful comments, J. Chun, T. Kailath, I. Koltracht and V. Rokhlin for sending me the copies of their papers and also Sally Goodall for her skill and patience in typing.

Bibliography

1. A. V. Aho, J. E. Hopcroft, and J. D. Ullman, *The design and analysis of computer algorithms*, Addison-Wesley, Reading, Mass., 1976.
2. G. S. Ammar and W. G. Gragg, *Superfast solution of real positive definite Toeplitz systems*, SIAM J. Matrix Anal. Appl. **9** (1988), 61–76.
3. C. R. Anderson, *A method of local corrections for computing the velocity field due to a distribution of vortex blobs*, J. Comput. Phys. **62** (1986), 111–123.
4. A. W. Appel, *An efficient program for many-body simulation*, SIAM J. Sci. Statist. Comput. **6** (1985), 85–103.
5. J. Barnes and P. Hut, *A hierarchical $O(N\log N)$ force-calculation algorithm*, Nature **324** (1986), 446–449.
6. R. R. Bitmead and B. D. O. Anderson, *Asymptotically fast solution of Toeplitz and related systems of linear equations*, Linear Algebra Appl. **34** (1980), 103–116.
7. A. Borodin and I. Munro, *The computational complexity of algebraic and numeric problems*, American Elsevier, New York, 1975.

8. J. F. Canny, E. Kaltofen, and Y. Lakshman, *Solving systems of non-linear polynomial equations faster*, Proc. ACM-SIGSAM Internat. Symposium on Symbolic and Algebraic Computations, ACM, New York, 1989, pp. 34–42.

9. J. Chun and T. Kailath, *Divide-and-conquer solution of least-squares problems for matrices with displacement structure*, SIAM J. Matrix Anal. Appl. (submitted).

10. J. Chun, T. Kailath, and H. Lev-Ari, *Fast parallel algorithm for QR-factorization of structured matrices*, SIAM J. Sci. Statist. Comput. **8** (1987), 899–913.

11. N. Gastinel, *Inversion d'une matrice généralisant la matrice de Hilbert*, Chiffres **3** (1960), 149–152.

12. A. Gerasoulis, *A fast algorithm for the multiplication of generalized Hilbert matrices with vectors*, Math. Comp. **50** (1987), 179–188.

13. I. Gohberg, T. Kailath, and I. Koltracht, *Efficient solution of linear systems of equations with recursive structure*, Linear Algebra Appl. **80** (1986), 81–113.

14. I. Gohberg, T. Kailath, I. Koltracht, and P. Lancaster, *Linear complexity parallel algorithms for linear systems of equations with recursive structure*, Linear Algebra Appl. **88/89** (1987), 271–315.

15. G. Heining and K. Rost, *Algebraic methods for Toeplitz-like matrices and operators*, Operator Theory, vol. 13, Birkhäuser, 1984.

16. T. Kailath and J. Chun, *Generalized Gohberg-Semencul formulas for matrix inversion*, Operator Theory: Advances and Applications, vol. 40, Birkhäuser, 1989, pp. 231–246.

17. T. Kailath, S.-Y. Kung, and M. Morf, *Displacement ranks of matrices and linear equations*, J. Math. Anal. Appl. **68** (1979), 395–407.

18. T. Kailath, A. Viera, and M. Morf, *Inverses of Toeplitz operators, innovations, and orthogonal polynomials*, SIAM Rev. **20** (1978), 106–119.

19. A. Leonard, *Vortex methods for flow simulation*, J. Comput. Phys. **37** (1980), 289–335.

20. B. R. Musicus, *Levinson and fast Choleski algorithms for Toeplitz and almost Toeplitz matrices*, Internal Report, Lab. of Electonics, M.I.T., 1981.

21. S. T. O'Donnell and V. Rokhlin, *A fast algorithm for numerical evaluation of conformal mappings*, Research Report RR-554, Yale Univ., Dept of Computer Science, 1987.

22. A. M. Odlyzko and A. Schönhage, *Fast algorithms for multiple evaluations of the Riemann zeta function*, Trans. Amer. Math. Soc. **309** (1988), 797–809.

23. V. Pan, *New effective methods for computations with structured matrices*, Technical Report 88-28, Computer Science Dept., SUNY Albany, 1988.

24. ——, *Fast and efficient parallel inversion of Toeplitz and block Toeplitz matrices*, Operator Theory: Advances and Applications, vol. 40, Birkhäuser, 1989, pp. 359–389.

25. ——, *Parallel least-squares solution of general and Toeplitz-like linear systems*, manuscript, 1990.

26. F. Rokhlin, *Rapid solution of integral equations of classical potential theory*, J. Comput. Phys. **60** (1985), 187–207.

27. V. Rokhlin, *A fast algorithm for the discrete Laplace transform*, J. Complexity **4** (1988), 12–32.

28. M. Tismenetsky, *Bésoutians, Toeplitz and Hankel matrices in the spectral theory of matrix polynomials*, Ph.D. thesis, Technion, Haifa, 1981.

Department of Mathematics, Lehman College, CUNY, Bronx, New York 10468

Department of Computer Science, SUNY Albany, Albany, New York 12222