Displacement Structure Approach to Polynomial Vandermonde and Related Matrices

Thomas Kailath and Vadim Olshevsky[1]

Information Systems Laboratory,
Stanford University,
Stanford CA 94305 – 4055, *U.S.A.*
E-mail : *tk@rascals.stanford.edu*
olshevsk@rascals.stanford.edu

ABSTRACT

In this paper we introduce a new class of what we shall call *polynomial Vandermonde-like* matrices. This class generalizes the polynomial Vandermonde matrices studied earlier by various authors, who derived explicit inversion formulas and fast algorithms for inversion and for solving the associated linear systems. A displacement structure approach allows us to carry over all these results to the wider class of polynomial Vandermonde-like matrices.

---

0. INTRODUCTION

0.1. Polynomial Vandermonde matrices. *Polynomial Vandermonde matrices* have the form

$$V_Q(x) = \begin{bmatrix} Q_0(x_1) & Q_1(x_1) & \cdots & Q_{n-1}(x_1) \\ Q_0(x_2) & Q_1(x_2) & \cdots & Q_{n-1}(x_2) \\ \vdots & \vdots & & \vdots \\ Q_0(x_n) & Q_1(x_n) & \cdots & Q_{n-1}(x_n) \end{bmatrix}, \qquad (x = (x_1, x_2, ..., x_n)), \tag{0.1}$$

with $Q = \{Q_0(x), Q_1(x), ..., Q_{n-1}(x)\}$, where $Q_k(x)$ is a polynomial of degree k. These matrices appear in theories of interpolation and approximation, in calculation of Gaussian quadrature and elsewhere, and they have been studied by several authors, see, e.g., [KarSz], [G], [MB], [VS], [Hig1], [Hig2], [RO], [CR], [GO2], [KO1], [BKO].

The most studied are the ordinary Vandermonde matrices, where Q is the monomial basis, $P = \{1, x, x^2, ..., x^{n-1}\}$. However ordinary Vandermonde matrices are extremely ill-conditioned; in fact the condition number of a real Vandermonde matrix grows exponentially with the size [GI], [Ty]. Therefore other choices for Q may be more attractive from the computational point of view. In fact, what we may call *three-term Vandermonde matrices*, in which the polynomials $Q_k(x)$ satisfy three-term recurrence relations, can be much better conditioned. For example when the $Q_k(x)$ are the Chebyshev polynomials of the first kind, $T_k(x) = \cos(k \arccos x)$, and the nodes $x_j = \cos(\frac{2\pi j}{n})$ are the zeros of $T_n(x)$, then $V_Q(x)$ is the Discrete Cosine Transform matrix, which is orthogonal and therefore perfectly conditioned. Analogously, when the $Q_k(x)$ are Chebyshev polynomials of the second kind $U_k(x) = \frac{\sin(k \arccos x)}{\sin(x)}$ and $x_j = \cos(\frac{2\pi j}{n})$, then $V_Q(x)$ is the (scaled) Discrete

[1]This work was supported in part by ARO contract DAAH04-96-1-0176 and also by DARPA contract F49620-95-1-0525. The views and conclusions contained in these documents are those of the author(s) and should not be interpreted as necessarily representing the official policies or endorsements, either expressed or implied, of the Advanced Research Projects Agency, the Army Research office, or the U.S. Government.

Sine Transform matrix. We may note that the conditioning of $V_Q(x)$, where Q stands for several classical families of polynomials orthogonal on a real interval, was studied in [G].

The polynomials $Q_k(x)$ in the three-term Vandermonde matrices are orthogonal with respect to a suitable inner product on a real interval, and therefore these matrices have been called *Vandermonde-like matrices involving orthogonal polynomials*, see, e.g., [G], [Hig1], [CR]. We however use the designation *three-term Vandermonde matrices*, because in the context of displacement structure theory the postfix *"like"* has a different meaning, which we shall use below.

Three-term Vandermonde matrices exhibit many nice properties, among which one can mention the existence of explicit inversion formulas and the availability of special algorithms that use the three-term Vandermonde structure for inversion and for solving the associated linear system. Table 1 lists these results, as well as further references.

Table 1.

	Inversion formula	Inversion algorithm	Algorithm for solving linear system
Vandermonde matrices	Well-known formula, see, e.g., its variants in [K1],[P],[Tr], [L],[HR],[GO5] and its φ-circulant analog [GO1]	Parker-Forney-Traub algorithm [P],[F], [Tr] (see also [GO5])	Björck-Pereyra algorithm [BP] (see also [GL]); and [HR],[GKK], [GKKL],[CK],[BiP],[BKO]
Chebyshev–Vandermonde matrices	Gohberg–Olshevsky formulas [CO2]	Gohberg–Olshevsky algorithm [GO2]	Reichel–Opfer algorithm [RO]; and Boros-Kailath-Olshevsky algorithm [BKO]
Three-term Vandermonde matrices	Verde–Star formula [VS] and Gohberg–Olshevsky formula [GO2]	Calvetti–Reichel algorithm [CR]	Heinig-Hoppe-Rost algorithm [HHR] and Higham algorithm [Hig1], [Hig2]

All the algorithms in Table 1 are called *fast algorithms*, because their complexity of $O(n^2)$ operations compares favorably with the $O(n^3)$ operations of general purpose algorithms like Gaussian elimination. Now inversion and fast solution of a linear system are two classical applications of the concept of displacement structure, see [KS2]. After briefly recalling some basic definitions of displacement structure theory, we shall use it to examine not only the polynomial Vandermonde matrices in (0.1), but also certain natural generalizations thereof, which we shall call *polynomial Vandermonde-like* matrices. We shall see that the concept allows us to nicely unify and extend the results in Table 1. The main reason is that the displacement structure is essentially preserved under inversion, see [KKM] and [KS2].

0.2. Displacement structure. Numerous applications give rise to matrices with a certain pattern of structure, e.g. to Toeplitz, Hankel, Toeplitz-plus-Hankel, polynomial Vandermonde, Cauchy, Pick matrices, and various others. Different kinds of structured matrices have many common properties; as a matter of fact, many results that hold for one pattern of structure have their counterparts for all other structured classes. The latter fact received an explanation in the framework of displacement structure theory. It turned out that the crucial common feature of all the above matrices is that they all have low displacement rank in the sense that it was introduced in [KKM] and later was much studied and generalized (see the reviews [K], [KS2]). To be more concrete, let us introduce the necessary notations.

Following [KS2], which contains results in the most general form, introduce in $\mathbf{C}^{n\times n}$ a linear *displacement operator* $\nabla_{\{\Omega,\Delta,F,A\}}(\cdot):\mathbf{C}^{n\times n}\to\mathbf{C}^{n\times n}$ that transforms each matrix $R\in\mathbf{C}^{n\times n}$ to its *displacement*, $\nabla_{\{\Omega,\Delta,F,A\}}(R)=\Omega\cdot R\cdot\Delta-F\cdot R\cdot A$, where $\Omega,\Delta,F,A\in\mathbf{C}^{n\times n}$ are given matrices. Let $\mathrm{rank}\nabla_{\{\Omega,\Delta,F,A\}}(R)=\alpha$; then one can factor (non-uniquely)

$$\nabla_{\{\Omega,\Delta,F,A\}}(R)=\Omega\cdot R\cdot\Delta-F\cdot R\cdot A=G\cdot B \qquad G\in\mathbf{C}^{n\times\alpha},B\in\mathbf{C}^{\alpha\times n} \tag{0.2}$$

The number α in (0.2) is called the $\{\Omega,\Delta,F,A\}$-*displacement rank* of R, and the pair of matrices $\{G,B\}$ on the right hand side of (0.2) is called a *minimal generator* of R. For example, it can easily be checked that shift-invariance property of a Toeplitz matrix $T=\left[\begin{array}{c} t_{i-j}\end{array}\right]_{1\le i,j\le n}$ implies that

$$T-Z_0\cdot T\cdot Z_0^T=\left[\begin{array}{cccc} t_0 & t_{-1} & \cdots & t_{-n+1}\\ t_1 & 0 & \cdots & 0\\ \vdots & \vdots & \ddots & \vdots\\ t_{n-1} & 0 & \cdots & 0\end{array}\right]=\left[\begin{array}{cc} \frac{t_0}{2} & 1\\ t_1 & 0\\ \vdots & \vdots\\ t_{n-1} & 0\end{array}\right]\cdot\left[\begin{array}{cccc} 1 & 0 & \cdots & 0\\ \frac{t_0}{2} & t_{-1} & \cdots & t_{-n+1}\end{array}\right], \tag{0.3}$$

where Z_0 stands for the lower shift matrix. In other words, (0.3) means that the $\{I,I,Z_0,Z_0^T\}$-displacement rank of an arbitrary Toeplitz matrix does not exceed 2. Clearly, inverses of Toeplitz matrices, products of Toeplitz matrices, mosaic Toeplitz matrices (which are also called Toeplitz-block matrices), are not Toeplitz matrices themselves. At the same time it has been observed that they all belong to the more general class of *Toeplitz-like* matrices, which were defined in [KKM] as matrices with small $\{I,I,Z_0,Z_0^T\}$-displacement rank. The displacement structure approach allowed to naturally carry over results from Toeplitz matrices to the wider class of Toeplitz-like matrices. In particular, the fact that the displacement rank of a matrix is (essentially) inherited by its inverse allowed to obtain explicit inversion formulas for Toeplitz-like matrices, generalizing the well known Gohberg–Semencul formula [GF] for inversion of Toeplitz matrices. Furthermore, the fact that the displacement rank of a matrix is inherited by its Schur complements also allowed to obtain a fast implementation of Gaussian elimination for Toeplitz-like matrices, generalizing the classical Schur algorithms for Hermitian Toeplitz matrices (see, e.g., [KS2]).

0.3. Vandermonde–like and Chebyshev–Vandermonde–like displacement structure. Already in the first "displacement" paper [KKM] it was noted that above approach is rather general and is not restricted to the Toeplitz-like choice $\nabla_{\{I,I,Z_0,Z_0^T\}}$ for the displacement operator. Confirming this expectation, it was very soon recognized (starting from [HR]) that not only Toeplitz, but also Hankel, Toeplitz-plus-Hankel, Vandermonde and Cauchy matrices have low displacement rank for appropriate choices of the matrices $\{\Omega,\Delta,F,A\}$ in (0.2). Moreover, arguments similar to those mentioned above justify the following designations for the choices $\Delta=F=I$ and for the indicated specific choices of Ω,A :

Table 2.

Vandermonde–like [HR]	$\Omega=\mathrm{diag}(x_1,...,x_n)$	$A=Z_0^T$
Cauchy–like [HR]	$\Omega=\mathrm{diag}(c_1,...,c_n)$	$A=\mathrm{diag}(d_1,...,d_n)$
Toeplitz–plus–Hankel–like [HJR], [GK1], [SLAK], [GKO]	$\Omega=Y_{00}$	$A=Y_{11}$

where as above Z_0 is the lower shift matrix and $Y_{\gamma,\delta}=Z_0+Z_0^T+\gamma e_1e_1^T+\delta e_ne_n^T$.

Furthermore, in a recent paper [KO1] it was observed that Chebyshev–Vandermonde matrices (i.e. matrices of the form (0.1) in which Q stands for Chebyshev polynomials) are also transformed to rank-one matrices by several displacement operators of the form (0.2). On this basis a more general class of *Chebyshev–Vandermonde-like* matrices was introduced as matrices with a low $\{\Omega,\Delta,F,A\}$-displacement rank, where $\Delta=F=I$ and Ω,A are as below :

Table 3.

Chebyshev–Vandermonde–like [KO1]	$\Omega = \mathrm{diag}(\frac{1}{x_1}, ..., \frac{1}{x_n})$	$A = 2 \cdot \sum_{i=1}^{[\frac{n}{2}]} (-1)^{i-1} Z_0^{2i-1}$
Chebyshev–Vandermonde–like [KO1]	$\Omega = \mathrm{diag}(x_1, ..., x_n)$	$A = Y_{\gamma,\delta}$

Moreover, the displacement structure approach allowed us to carry over many results from Vandermonde and Chebyshev–Vandermonde matrices to the wider classes of Vandermonde-like and Chebyshev–Vandermonde–like matrices, respectively. A list of these generalizations and further references are given in the Table 4.

Table 4.

	Inversion formula	Inversion algorithm	Algorithm for solving linear system
Vandermonde–like matrices	Heinig-Rost formula [HR] Gohberg-Olshevsky formulas [GO3], [GO5]	Gohberg-Olshevsky algorithm [GO5]	[HR], [CK] ,[KS1], [KS2], [GO3],[GKO]
Chebyshev–Vandermonde–like matrices	Kailath–Olshevsky formula [KO1]	Kailath–Olshevsky algorithm [KO1]	Kailath–Olshevsky algorithm [KO1]

All the algorithms in Table 4 have complexity $O(\alpha n^2)$ operations, where α is the displacement rank of a matrix.

0.5. Main results. Comparing Tables 1 and 2 immediately gives rise to the following two questions. First, is there a quadruple of matrices Ω, Δ, F, A, such that the three-term Vandermonde and the polynomial Vandermonde matrices have low $\{\Omega, \Delta, F, A\}$-displacement rank? In that case a new class of polynomial Vandermonde-like matrices can be associated with the corresponding displacement operator $\nabla_{\{\Omega,\Delta,F,A\}}(\cdot)$. Second, can the results in the bottom line of Table 1 be carried over to this wider class of polynomial Vandermonde-like matrices?

The present paper presents affirmative answers to both these questions. More precisely, we suggest three alternative choices of matrices $\{\Omega, \Delta, F, A\}$, such that polynomial Vandermonde matrices are transformed by $\nabla_{\{\Omega,\Delta,F,A\}}(\cdot)$ as in (0.2) to rank-one matrices. Each of these displacement operators can be used to define a new class of matrices with low $\{\Omega, \Delta, F, A\}$-displacement rank. However, a natural question is how these three new classes of matrices relate to each other? We show that no matter which of the three displacement operators is chosen, all three definitions lead in fact to the same class of matrices, which we shall call the class of *polynomial Vandermonde-like matrices*.

Furthermore we show that all the results in the bottom line of Table 1 can be carried over to this wider class of matrices. In particular, we derive (a) two inversion formulas; (b) a structured implementation of Gaussian elimination with partial pivoting, and (c) an inversion algorithm for polynomial Vandermonde-like matrices. In a general situation, where the polynomial system Q in (0.1) is arbitrary, the complexity of the two latter algorithms is $O(\alpha n^3)$ arithmetic operations, where α is the displacement rank. However when the polynomials in Q satisfy m-term recurrence relations, then this complexity reduces to $O(\alpha m n^2)$ operations (here we may remark that this result is new, not only for general polynomial Vandermonde-*like* matrices, but also for m-term Vandermonde matrices of the form (0.1)). Finally note that for the important special case in which the polynomials in Q satisfy three-term recurrence relations, all the algorithms proposed in the present paper have complexity $O(\alpha n^2)$.

0.6. Contents. In the first section we introduce three displacement operators associated with polynomial Vandermonde matrices. In section 2 we show that all these operators define the same class, the class of polynomial Vadndermonde-like matrices. Then in section 3 we introduce generalized associated polynomials that will be later used to describe the structure of inverses of polynomial Vandermonde-like matrices. In section 4 we list several necessary auxiliary properties. In section 5 we derive two inversion formulas for polynomial Vandermonde-like matrices. In sections 6 and 7 we design efficient algorithms for solving the associated linear system and for inversion. In section 8 we show how to transform polynomial Vandermonde-like matrices to Cauchy-like matrices, a fact that enables other interesting connections.

1. THREE DISPLACEMENT OPERATORS

Let $Q = \{Q_0(x), Q_1(x), ..., Q_n(x)\}$ be a system of $n+1$ polynomials satisfying the recurrence relations

$$Q_0(x) = \alpha_0,$$
$$Q_k(x) = \alpha_k \cdot x \cdot Q_{k-1}(x) - a_{k-1,k} \cdot Q_{k-1}(x) - a_{k-2,k} \cdot Q_{k-2}(x) - ... - a_{0,k} \cdot Q_0(x), \tag{1.1}$$

where $\alpha_k \neq 0$. These relations will allow us to write down three displacement operators $\nabla_{\{\Omega,A\}}(\cdot) : \mathbf{C}^{n\times n} \to \mathbf{C}^{n\times n}$ of the form

$$\nabla_{\{\Omega,\Delta,F,A\}}(R) = \Omega \cdot R \cdot \Delta - F \cdot R \cdot A \tag{1.2}$$

that will transform a polynomial Vandermonde matrix $V_Q(x)$ to rank-one matrices.

One of these operators will have a general form as in (1.2), while the other two will be of a more simple (Sylvester) form

$$\nabla_{\{\Omega,I,I,A\}}(R) = \Omega \cdot R - R \cdot A. \tag{1.3}$$

1.1. First displacement operator. We first construct from the coefficients α_k and $a_{i,k}$ in (1.1) the matrices

$$M_Q = \begin{bmatrix} 1 & a_{0,1} & a_{0,2} & \cdots & a_{0,n-1} \\ 0 & 1 & a_{1,2} & \cdots & a_{1,n-1} \\ \vdots & \ddots & \ddots & & \vdots \\ \vdots & & \ddots & \ddots & a_{n-2,n-1} \\ 0 & \cdots & \cdots & 0 & 1 \end{bmatrix}, \qquad N_Q = \begin{bmatrix} 0 & \alpha_1 & 0 & \cdots & 0 \\ 0 & 0 & \alpha_2 & \ddots & \ddots \\ 0 & 0 & 0 & \ddots & 0 \\ \vdots & & & \ddots & \alpha_{n-1} \\ 0 & \cdots & \cdots & 0 & 0 \end{bmatrix}. \tag{1.4}$$

Then the following statement holds.

LEMMA 1.1 *Let the polynomials* $Q = \{Q_0(x), Q_1(x), ..., Q_n(x)\}$ *be defined by* (1.1), *the matrices* M_Q *and* N_Q *be given by* (1.4) *and let* $D_x = \mathrm{diag}(x_1, x_2, ..., x_n)$. *Then the polynomial Vandermonde matrix* $V_Q(x)$ *satisfies*

$$\nabla_{\{I,M_Q,D_x,N_Q\}}(V_Q(x)) = V_Q(x) \cdot M_Q - D_x \cdot V_Q(x) \cdot N_Q = \begin{bmatrix} 1 \\ 1 \\ \vdots \\ 1 \end{bmatrix} \cdot \begin{bmatrix} \alpha_0 & 0 & \cdots & 0 \end{bmatrix}. \tag{1.5}$$

Proof. From the recurrence relations (1.1) it immediately follows that only the entries in the first column of the matrix $\nabla_{\{I,M_Q,D_x,N_Q\}}(V_Q(x))$ may differ from zero. Computing these entries leads to (1.5).

□

1.2. Second displacement operator. Let the polynomials in $Q = \{Q_0(x), Q_1(x), ..., Q_n(x)\}$ satisfy the recurrence relations (1.1). Following [MB], define for a polynomial

$$\Theta(x) = \theta_0 \cdot Q_0(x) + \theta_1 \cdot Q_1(x) + ... + \theta_{n-1} \cdot Q_{n-1}(x) + \theta_n \cdot Q_n(x),$$

its *confederate matrix*

$$C_Q(\Theta) = \begin{bmatrix} \frac{a_{01}}{\alpha_1} & \frac{a_{02}}{\alpha_2} & \frac{a_{03}}{\alpha_3} & \cdots & \cdots & \frac{a_{0,n}}{\alpha_n} - \frac{1}{\alpha_n} \cdot \frac{\theta_0}{\theta_n} \\ \frac{1}{\alpha_1} & \frac{a_{12}}{\alpha_2} & \frac{a_{13}}{\alpha_3} & \cdots & \cdots & \frac{a_{1,n}}{\alpha_n} - \frac{1}{\alpha_n} \cdot \frac{\theta_1}{\theta_n} \\ 0 & \frac{1}{\alpha_2} & \frac{a_{23}}{\alpha_3} & \cdots & \cdots & \frac{a_{2,n}}{\alpha_n} - \frac{1}{\alpha_n} \cdot \frac{\theta_2}{\theta_n} \\ 0 & 0 & \ddots & \ddots & & \vdots \\ \vdots & \vdots & \ddots & \ddots & \ddots & \vdots \\ 0 & 0 & \cdots & 0 & \frac{1}{\alpha_{n-1}} & \frac{a_{n-1,n}}{\alpha_n} - \frac{1}{\alpha_n} \cdot \frac{\theta_{n-1}}{\theta_n} \end{bmatrix} \tag{1.6}$$

with respect to the system Q. We refer to [MB] for many useful properties of the confederate matrix and only recall here that $\det(\lambda I - C_Q(\Theta)) = \Theta(x)/(\alpha_0 \cdot \alpha_1 \cdot ... \cdot \alpha_n \cdot \theta_n)$.

In this paper we shall pay special attention to the important case where the polynomials $Q_k(x)$ satisfy three-term recurrence relations of the form

$$Q_k(x) = (\alpha_k \cdot x - \beta_k) \cdot Q_{k-1}(x) - \gamma_k \cdot Q_{k-2}(x).$$

In this case the confederate matrix is of the almost tridiagonal form

$$C_Q(\Theta) = \begin{bmatrix} \frac{\beta_1}{\alpha_1} & \frac{\gamma_2}{\alpha_2} & 0 & \cdots & 0 & -\frac{1}{\alpha_n} \cdot \frac{\theta_0}{\theta_n} \\ \frac{1}{\alpha_1} & \frac{\beta_2}{\alpha_2} & \frac{\gamma_3}{\alpha_3} & \ddots & \vdots & -\frac{1}{\alpha_n} \cdot \frac{\theta_1}{\theta_n} \\ 0 & \frac{1}{\alpha_2} & \frac{\beta_3}{\alpha_3} & \ddots & 0 & \vdots \\ \vdots & 0 & \frac{1}{\alpha_3} & \ddots & \frac{\gamma_{n-1}}{\alpha_{n-1}} & -\frac{1}{\alpha_n} \cdot \frac{\theta_{n-3}}{\theta_n} \\ \vdots & \vdots & \ddots & \ddots & \frac{\beta_{n-1}}{\alpha_{n-1}} & \frac{\gamma_n}{\alpha_n} - \frac{1}{\alpha_n} \cdot \frac{\theta_{n-2}}{\theta_n} \\ 0 & 0 & \cdots & 0 & \frac{1}{\alpha_{n-1}} & \frac{\beta_n}{\alpha_n} - \frac{1}{\alpha_n} \cdot \frac{\theta_{n-1}}{\theta_n} \end{bmatrix},$$

which has been called a *comrade matrix* in [B], [MB]. Note also that in the simplest case of the power basis $P = \{1, x, x^2, ..., x^n\}$, $C_P(\Theta)$ reduces to the well known *companion matrix*

$$C_P(\Theta) = \begin{bmatrix} 0 & 0 & \cdots & 0 & -\frac{\theta_0}{\theta_n} \\ 1 & 0 & \cdots & 0 & -\frac{\theta_1}{\theta_n} \\ 0 & 1 & \ddots & \vdots & \vdots \\ \vdots & & \ddots & 0 & \vdots \\ 0 & \cdots & 0 & 1 & -\frac{\theta_{n-1}}{\theta_n} \end{bmatrix}.$$

The second displacement operator for $V_Q(x)$, described in the next lemma, involves the corresponding confederate matrix.

LEMMA 1.2 *Let polynomials $Q = \{Q_0(x), Q_1(x), ..., Q_n(x)\}$ be defined by* (1.1), *polynomial $\Theta(x)$ be arbitrary, and let $C_Q(\Theta)$ be the confederate matrix as in* (1.6) *of $\Theta(x)$ with respect to the system Q. Then*

$$\nabla_{\{D_x, I, I, C_Q(\Theta)\}}(V_Q(x)) = D_x \cdot V_Q(x) - V_Q(x) \cdot C_Q(\Theta) = \begin{bmatrix} \Theta(x_1) \\ \Theta(x_2) \\ \vdots \\ \Theta(x_n) \end{bmatrix} \cdot \begin{bmatrix} 0 & \cdots & 0 & \frac{1}{\alpha_n \cdot \theta_n} \end{bmatrix}. \tag{1.7}$$

In particular, if $x_1, x_2, ..., x_n$ are the n zeros of the polynomial $\Theta(x)$, then

$$V_Q(x) \cdot C_Q(\Theta) \cdot V_Q(x)^{-1} = D_x. \tag{1.8}$$

Lemma 1.2 is also deduced from the recurrence relations (1.1) without any difficulties.

□

We note that the formula (1.8) appeared earlier in [MB].

1.3. Third displacement operator. The displacement operator on the left hand side of (1.7) has the Sylvester form (1.3), with diagonal matrix $\Omega = D_x$ and Hessenberg matrix $A = C_Q(\Theta)$. To devise in Sec. 6 a fast implementation of Gaussian elimination with partial pivoting for $V_Q(x)$ we shall need one more displacement operator of Sylvester form (1.3) in which Ω is also a diagonal matrix, while $A = W_Q$ is an upper triangular matrix of the form

$$W_Q = \begin{bmatrix} 0 & w_{12} & w_{13} & \cdots & w_{1n} \\ 0 & 0 & w_{23} & \ddots & \vdots \\ \vdots & & \ddots & \ddots & w_{n-2,n} \\ \vdots & & & \ddots & w_{n-1,n} \\ 0 & \cdots & \cdots & \cdots & 0 \end{bmatrix}, \tag{1.9}$$

whose entries are specified by

$$Q_1(x) = \delta_1 \cdot Q_0(x) + x \cdot w_{12} Q_0(x),$$

$$Q_k(x) = \delta_k \cdot Q_0(x) + x \cdot \sum_{i=1}^{k} w_{i,k+1} \cdot Q_{i-1}(x) \qquad (k = 2, 3, ..., n-1). \tag{1.10}$$

Since $\deg Q_k(x) = k$, the polynomials $\{Q_0(x),\ x \cdot Q_0(x),\ x \cdot Q_1(x),\ ...\ ,\ x \cdot Q_{n-1}(x)\}$ form a basis in the linear space $\mathbf{C}_k[x]$ of all polynomials of degree not exceeding n. Hence the numbers δ_k and $w_{i,j}$ ($j > i$) are uniquely determined by (1.10) and the matrix W_Q in (1.9) is well defined. The following lemma immediately follows from the definitions and it allows an efficient computation of the entries of W_Q.

Lemma 1.3 *Let numbers $\delta_1, ..., \delta_{n-1}$ and matrix $W_Q = \begin{bmatrix} w_{i,j} \end{bmatrix}_{1 \le i,j \le n}$ be specified by* (1.9), (1.10). *Denote by $\mathbf{w_k} \in \mathbf{C}^n$ the k-th column of $W_Q = \begin{bmatrix} \mathbf{w}_1 & \mathbf{w}_2 & \cdots & \mathbf{w}_n \end{bmatrix}$. Then $w_{12} = \alpha_1$, and*

$$\begin{bmatrix} \delta_k \\ \mathbf{w}_{k+1} \end{bmatrix} = \alpha_k \cdot \begin{bmatrix} 0 \\ \vdots \\ 0 \\ 1 \end{bmatrix} - a_{k-1,k} \cdot \begin{bmatrix} \delta_{k-1} \\ \mathbf{w}_k \end{bmatrix} - a_{k-2,k} \cdot \begin{bmatrix} \delta_{k-2} \\ \mathbf{w}_{k-1} \end{bmatrix} - ... - a_{1,k} \cdot \begin{bmatrix} \delta_1 \\ \mathbf{w}_2 \end{bmatrix} - a_{0,k} \cdot \begin{bmatrix} 1 \\ \mathbf{0} \end{bmatrix}. \tag{1.11}$$

The following lemma follows easily from (1.10), and it uses W_Q to describe one more kind of displacement equation for $V_Q(x)$.

Lemma 1.4 *Let $x_1, x_2, ..., x_n$ be n nonzero numbers,*

$$\Omega = D_{\frac{1}{x}} = \mathrm{diag}(\frac{1}{x_1}, \frac{1}{x_2}, ... \frac{1}{x_n}), \tag{1.12}$$

and let the matrix $W_Q \in \mathbf{C}^{n \times n}$ be defined as in (1.9). *Then*

$$\nabla_{\{D_{\frac{1}{x}}, I, I, W_Q\}}(V_Q(x)) = D_{\frac{1}{x}} \cdot V_Q(x) - V_Q(x) \cdot W_Q = \begin{bmatrix} \frac{1}{x_1} \\ \frac{1}{x_2} \\ \vdots \\ \frac{1}{x_n} \end{bmatrix} \cdot \begin{bmatrix} \alpha_0 & \alpha_0\delta_1 & \cdots & \alpha_0\delta_{n-1} \end{bmatrix}. \tag{1.13}$$

1.4. Examples. To show the unifying nature of the displacement concept, we conclude this section with several examples where the displacement operators (1.5), (1.7) and (1.13) are specialized to yield the (ordinary) Vandermonde and Chebyshev–Vandermonde matrices. Some of these examples can be found in [HR], [GO3], [KO1].

EXAMPLE 1.5 *Let P stand for the power basis $P = \{1, x, ..., x^{n-1}\}$, i.e. $\alpha_k = 1$, $a_{i,k} = 0$ for $i = 0, 1, ..., n-1$; $k = 1, 2, ..., n$. For a Vandermonde matrix $V_P(x)$ the displacement equations* (1.5), (1.7) *and* (1.13), *respectively, have the forms*

$$V_P(x) - D_x \cdot V_P(x) \cdot Z_0^T = \begin{bmatrix} 1 \\ 1 \\ \vdots \\ 1 \end{bmatrix} \cdot \begin{bmatrix} 1 & 0 & \cdots & 0 \end{bmatrix},$$

$$D_x \cdot V_P(x) - V_P(x) \cdot C_P(\Theta) = \begin{bmatrix} \Theta(x_1) \\ \Theta(x_2) \\ \vdots \\ \Theta(x_n) \end{bmatrix} \cdot \begin{bmatrix} 0 & \cdots & 0 & \frac{1}{\theta_n} \end{bmatrix},$$

where $C_P(\Theta)$ is a companion matrix of an arbitrary polynomial $\Theta(x)$, and

$$D_{\frac{1}{x}} \cdot V_P(x) - V_P(x) \cdot Z_0^T = \begin{bmatrix} \frac{1}{x_1} \\ \frac{1}{x_2} \\ \vdots \\ \frac{1}{x_n} \end{bmatrix} \cdot \begin{bmatrix} 1 & 0 & \cdots & 0 \end{bmatrix}. \tag{1.14}$$

EXAMPLE 1.6 *Let U stand for the basis $U = \{U_0(x), U_1(x), ..., U_{n-1}(x)\}$ of Chebyshev polynomials of the second kind, i.e. the relations* (1.1) *became*

$$U_0(x) = 1; \qquad U_1(x) = 2x; \qquad U_k(x) = 2x \cdot U_{k-1}(x) - U_{k-2}(x).$$

For a Chebyshev–Vandermonde matrix $V_U(x)$, the displacement equations (1.5) (1.7) *and* (1.13) *have the forms*

$$V_U(x) \cdot (I + (Z_0^T)^2) - 2D_x \cdot V_U(x) \cdot Z_0^T = \begin{bmatrix} 1 \\ 1 \\ \vdots \\ 1 \end{bmatrix} \cdot \begin{bmatrix} 1 & 0 & \cdots & 0 \end{bmatrix},$$

$$D_x \cdot V_U(x) - V_U(x) \cdot \frac{1}{2}(C_U(\Theta) + Z_0^T) = \begin{bmatrix} \Theta(x_1) \\ \Theta(x_2) \\ \vdots \\ \Theta(x_n) \end{bmatrix} \cdot \begin{bmatrix} 0 & \cdots & 0 & \frac{1}{\theta_n} \end{bmatrix},$$

where $C_U(\Theta)$ is the comrade matrix of an arbitrary polynomial $\Theta(x)$ with respect to the system U, and

$$D_{\frac{1}{x}} \cdot V_U(x) - V_U(x) \cdot 2\sum_{i=1}^{[\frac{n}{2}]}(-1)^{i-1} \cdot (Z_0^T)^{2i-1} = \begin{bmatrix} \frac{1}{x_1} \\ \frac{1}{x_2} \\ \vdots \\ \frac{1}{x_n} \end{bmatrix} \cdot \begin{bmatrix} 1 & 0 & -1 & 0 & 1 & 0 & -1 & \cdots \end{bmatrix}.$$

$\square$

2. POLYNOMIAL VANDERMONDE-LIKE MATRICES

Lemmas 1.1 - 1.3 claim that a polynomial Vandermonde matrix $V_Q(x)$ is transformed to a rank-one matrix by each of the displacement operators $\nabla_{\{\Omega,\Delta,F,A\}}$ in (1.5), (1.7) or (1.13). Any of these displacement operators can be chosen for defining a more general class of matrices with a low $\{\Omega,\Delta,F,A\}$-displacement rank. The immediate question is how these three classes of matrices will relate to each other. In this section we shall show that if a square matrix has *low* displacement rank with respect to any one of the above displacement operators, then it also has low displacement rank with respect to the other two. This means that no matter which of the displacement operators in (1.5), (1.7) or (1.13) is chosen, all three definitions describe in fact the same class of matrices, which we shall call the class of *polynomial Vandermonde–like* matrices.

THEOREM 2.1 *Let* $\nabla_{\{I,M_Q,D_x,N_Q\}}(\cdot)$ *and* $\nabla_{\{D_x,I,I,C_Q(\Theta)\}}(\cdot)$ *be the first and the second displacement operators given by* (1.5) *and* (1.7), *respectively. Then for any matrix* $R \in \mathbf{C}^{n\times n}$ *we have*

$$|\mathrm{rank}\nabla_{\{I,M_Q,D_x,N_Q\}}(R) - \mathrm{rank}\nabla_{\{D_x,I,I,C_Q(\Theta)\}}(R)| < 2. \tag{2.1}$$

Proof. Denote by Z_1 a circulant lower shift matrix and note that the matrix

$$N_1 = \begin{bmatrix} 0 & \alpha_1 & 0 & \cdots & 0 \\ 0 & 0 & \alpha_2 & \ddots & \ddots \\ 0 & 0 & 0 & \ddots & 0 \\ \vdots & & \ddots & \ddots & \alpha_{n-1} \\ \alpha_n & \cdots & \cdots & 0 & 0 \end{bmatrix} = \mathrm{diag}(\alpha_1,\alpha_2,...,\alpha_n)\cdot Z_1^T$$

is a rank-one perturbation of the matrix N_Q in (1.4). Hence the rank of the matrix

$$\nabla_{\{I,M_Q,D_x,N_1\}}(R) = R\cdot M_Q - D_x\cdot R\cdot N_1 \tag{2.2}$$

may differ from the $\nabla_{\{I,M_Q,D_x,N_Q\}}$-displacement rank of R by no more than one. Hence multiplying (2.2) by N_1^{-1} from the right, one obtains

$$|\mathrm{rank}\nabla_{\{D_x,I,I,M_Q\cdot N_1^{-1}\}}(R) - \mathrm{rank}\nabla_{\{I,M_Q,D_x,N_Q\}}(R)| < 1. \tag{2.3}$$

Furthermore, it easy to check that $M_Q\cdot N_1^{-1} = M_Q\cdot Z_1\cdot\mathrm{diag}(\frac{1}{\alpha_1},\frac{1}{\alpha_2},...,\frac{1}{\alpha_n})$ differs from the matrix $C_Q(\Theta)$ in (1.6) only in the entries of the last column, i.e. matrix $M_Q\cdot N_1^{-1}$ is a rank-one perturbation of $C_Q(\Theta)$. Therefore (2.3) implies (2.1) and the assertions of the Theorem follow.

$\square$

From Theorem 2.1 it follows that the two displacement operators in (1.5) and (1.7) can be used to define polynomial Vandermonde-like matrices as matrices with low displacement rank; however the actual displacement rank of a matrix will depend on the displacement operator actually used. The next statement shows that the displacement operator in (1.13) is also associated with the same class of polynomial Vandermonde-like matrices.

THEOREM 2.2 *Let* $\nabla_{\{I,M_Q,D_x,N_Q\}}(\cdot)$ *and* $\nabla_{\{D_{\frac{1}{x}},I,I,W_Q\}}(\cdot)$ *be the second and the third displacement operators defined by* (1.7) *and by* (1.13), *respectively. Then for an arbitrary matrix* $R \in \mathbf{C}^{n\times n}$ *we have*

$$\mathrm{rank}\nabla_{\{I,M_Q,D_x,N_Q\}}(R) = \mathrm{rank}\nabla_{\{D_{\frac{1}{x}},I,I,W_Q\}}(R). \tag{2.4}$$

This theorem immediately follows from the next auxiliary statement, which will also be used later.

Lemma 2.3 *The matrix W_Q specified by* (1.9) *and* (1.10) *admits the representation*

$$W_Q = N_Q \cdot M_Q^{-1}, \tag{2.5}$$

where M_Q and N_Q are given by (1.3).

Proof. By adding the first displacement equation (1.5), multiplied by M_Q^{-1} from the right to the third displacement equation (1.13), multiplied by D_x from the left, we obtrain :

$$D_x \cdot V_Q(x) \cdot [W_Q - N_Q \cdot M_Q^{-1}] = \alpha_0 \begin{bmatrix} 1 \\ \vdots \\ 1 \end{bmatrix} \cdot (\begin{bmatrix} 1 & \delta_1 & \cdots & \delta_{n-1} \end{bmatrix} - \begin{bmatrix} 1 & 0 & \cdots & 0 \end{bmatrix} \cdot M_Q^{-1}).$$

The entries in the first row of the matrix on the left hand side are polynomials in x_1, vanishing at zero, whereas the entries in the first row of the matrix on the right hand side are constants. Since x_k is arbitrary, both matrices are identically zero, and (2.5) follows.

□

3. GENERALIZED ASSOCIATED POLYNOMIALS

Ordinary polynomial Vandermonde matrices $V_P(x)$ (i.e., where P is the monomial basis) can be rapidly inverted in $O(n^2)$ arithmetic operations, and the fast algorithm for that purpose was independently derived by Parker [P], Forney [F], Traub [Tr], and several others. In particular, Traub derived this algorithm by using the properties of the so-called *associated polynomials* (or Horner polynomials), which describe the structure of the inverse Vandermonde matrix. In this section we describe several useful properties of the so-called *generalized associated polynomials* $\hat{Q} = \{\hat{Q}_0(x), \hat{Q}_1(x), ..., \hat{Q}_n(x)\}$. These are associated with a given system $Q = \{Q_0(x), Q_1(x), ..., Q_n(x)\}$ via

$$\hat{Q}_n(x) = Q_n(x), \qquad \text{and} \qquad \frac{Q_n(x) - Q_n(y)}{x - y} = \sum_{i=0}^{n-1} Q_i(x) \cdot \hat{Q}_{n-1-i}(y).$$

In fact, the $\hat{Q}$ determine the structure of $V_Q(x)^{-1}$, thus leading to an efficient inversion algorithm. However, because of space limitations we do not pursue these connections with the original Parker-Forney-Traub algorithm here; this issue is addressed in a parallel contribution [KO2]. In this paper our goal is to address a more general problem, and to exploit the properties of the generalized associated polynomials to derive inversion formulas for polynomial Vandermonde-*like* matrices in Section 4, and an algorithm for inversion of polynomial Vandermonde-*like* matrices in Section 6. To this end we shall need several useful properties of generalized of associated polynomials, which will be introduced via the equivalent (see, e.g., [KO2] for the details) definition :

$$\hat{Q}_0(x) = \hat{\alpha}_0,$$
$$\hat{Q}_k(x) = \hat{\alpha}_k \cdot x \cdot \hat{Q}_{k-1}(x) - \hat{a}_{k-1,k} \cdot \hat{Q}_{k-1}(x) - \hat{a}_{k-2,k} \cdot \hat{Q}_{k-2}(x) - ... - \hat{a}_{0,k} \cdot \hat{Q}_0(x). \tag{3.1}$$

where

$$\hat{\alpha}_k = \alpha_{n-k}, \qquad (k = 0, 1, ..., n),$$

and

$$\hat{a}_{k,j} = \frac{\alpha_{n-j}}{\alpha_{n-k}} a_{n-j,n-k} \qquad (k = 0, 1, ...n-1; j = 1, 2, ..., n). \tag{3.2}$$

These relations can be conveniently reformulated in matrix form as follows.

LEMMA 3.1 *Let* $Q_k(x)$ *and* $\hat{Q}_k(x)$ *be the systems of polynomials, specified by* (1.1) *and* (3.1), *resp. Then the following statements hold :*

(i) $Q_n(x) = \hat{Q}_n(x)$.

(ii) *The confederate matrices* $C_Q(Q_n)$ *and* $C_{\hat{Q}}(Q_n)$ *are related by*

$$C_{\hat{Q}}(Q_n) = \tilde{I} \cdot C_Q(Q_n)^T \cdot \tilde{I}, \tag{3.3}$$

where $\tilde{I}$ *stands for the anti-diagonal identity matrix.*

(iii) *Finally,*

$$W_{\hat{Q}} = \tilde{I} \cdot W_Q^T \cdot \tilde{I}, \tag{3.4}$$

where W_Q *is the matrix defined by* (1.9), (1.10).

Proof. Using the definition of $C_Q(\Theta)$ in (1.6) and then (3.2), it is straightforward to check that the confederate matrices $C_{\hat{Q}}(\hat{Q}_n)$ and $C_Q(Q_n)$ are related as

$$C_{\hat{Q}}(\hat{Q}_n) = \tilde{I} \cdot C_Q(Q_n)^T \cdot \tilde{I}.$$

Since $\det(C_Q(\Theta)) = \Theta(x)/(\alpha_0 \cdot ... \cdot \alpha_n \cdot \theta_n)$, from the latter equality it follows that the polynomials $Q_n(x)$ and $\hat{Q}_n(x)$ share the same zeros. Furthermore from (3.1) it follows that the number $\alpha_0 \cdot \alpha_1 \cdot ... \cdot \alpha_n \cdot \theta_n$ is the leading coefficient of both polynomials, and assertions (i) and (ii) follow.

In order to prove (3.4) let us recall that by (2.5),

$$W_Q = Z_0^T \cdot (M_Q \cdot \mathrm{diag}(\frac{1}{\alpha_0}, \frac{1}{\alpha_1}, ..., \frac{1}{\alpha_{n-1}}))^{-1}, \tag{3.5}$$

where M_Q is defined as in (1.4). Therefore

$$\tilde{I} \cdot W_Q^T \cdot \tilde{I} = (\tilde{I} \cdot \mathrm{diag}(\frac{1}{\alpha_0}, \frac{1}{\alpha_1}, ..., \frac{1}{\alpha_{n-1}}) \cdot M_Q^T \cdot \tilde{I})^{-1} \cdot Z_0^T =$$

$$\begin{bmatrix} \frac{1}{\alpha_{n-1}} & \frac{a_{n-2,n-1}}{\alpha_{n-1}} & \cdots & \frac{a_{1,n-1}}{\alpha_{n-1}} & \frac{a_{0,n-1}}{\alpha_{n-1}} \\ 0 & \frac{1}{\alpha_{n-2}} & \cdots & \frac{a_{1,n-2}}{\alpha_{n-2}} & \frac{a_{0,n-2}}{\alpha_{n-2}} \\ \vdots & & \ddots & \vdots & \vdots \\ \vdots & & \ddots & \frac{1}{\alpha_1} & \frac{a_{0,1}}{\alpha_1} \\ 0 & \cdots & \cdots & 0 & \frac{1}{\alpha_0} \end{bmatrix}^{-1} \cdot Z_0^T = Z_0^T \cdot \begin{bmatrix} \frac{1}{\alpha_n} & \frac{a_{n-1,n}}{\alpha_n} & \cdots & \frac{a_{2,n}}{\alpha_n} & \frac{a_{1,n}}{\alpha_n} \\ 0 & \frac{1}{\alpha_{n-1}} & \cdots & \frac{a_{2,n-1}}{\alpha_{n-1}} & \frac{a_{1,n-1}}{\alpha_{n-1}} \\ \vdots & & \ddots & \vdots & \vdots \\ \vdots & & \ddots & \frac{1}{\alpha_2} & \frac{a_{1,2}}{\alpha_2} \\ 0 & \cdots & \cdots & 0 & \frac{1}{\alpha_1} \end{bmatrix}^{-1} =$$

$$= Z_0^T \cdot (M_{\hat{Q}} \cdot \mathrm{diag}(\frac{1}{\hat{\alpha}_0}, \frac{1}{\hat{\alpha}_1}, ..., \frac{1}{\hat{\alpha}_{n-1}}))^{-1}.$$

The latter equality and (3.5) imply (3.4).

□

As we shall see in section 5 below, the associated polynomials define the structure of the inverses of polynomial Vandermonde-like matrices. However, some necessary auxiliary results are provided first.

4. CHANGE OF BASIS

4.1. Passing to another polynomial basis. Along with the polynomials $Q = \{Q_1(x), Q_2(x), ..., Q_n(x)\}$ defined by (1.1), consider another set of polynomials $R = \{R_1(x), R_2(x), ..., R_n(x)\}$, defined by analogous recurrences :

$$R_0(x) = \beta_0,$$
$$R_k(x) = \beta_k \cdot x \cdot R_{k-1}(x) - b_{k-1,k} \cdot R_{k-1}(x) - b_{k-2,k} \cdot R_{k-2}(x) - ... - b_{0,k} \cdot R_0(x), \tag{4.1}$$

where $\beta_k \neq 0$. Let

$$S_{RQ} = \left[\begin{array}{c} s_{ij} \end{array}\right]_{1 \leq i,j \leq n} \tag{4.2}$$

be the matrix corresponding to passing from the basis Q to the basis R in the linear space $C_n[x]$ of all polynomials whose degree does not exceed n. Clearly, the entries of the upper triangular matrix S_{RQ} are specified by

$$Q_k(x) = \sum_{i=0}^{k} s_{i+1,k+1} \cdot R_i(x), \tag{4.3}$$

and therefore

$$V_Q(x) = V_R(x) \cdot S_{RQ}. \tag{4.4}$$

4.2 Change of the confederate matrix. The following lemma of [MB] shows how the confederate matrix changes under the passagge to another polynomial basis.

LEMMA 4.1 (MB) *Let S_{RQ} be an upper truangular matrix, defined in (4.2). Then*

$$C_Q(\Theta) = S_{RQ}^{-1} \cdot C_R(\Theta) \cdot S_{RQ}. \tag{4.5}$$

For the simplest case, where R stands for the power basis $P = \{1, x, x^2, ..., x^n\}$ and $\Theta(x) = x^n$, we have

$$C_Q(x^n) = S_{PQ}^{-1} \cdot Z_0 \cdot S_{PQ}.$$

4.3. Computing the entries of S_{RQ}. The above lemma allows efficient computation of the columns of S_{RQ} by using the same recurrence relations (1.1), in which multiplication by x is replaced with the multiplication by the confederate matrix $C_R(\Theta)$. Moreover the following statement holds.

COROLLARY 4.2 *Let the systems of polynomials Q and R be given by* (1.1) *and* (4.1), *respectively. Then the columns $\mathbf{s}_k \in \mathbf{C}^n$ of the upper triangular matrix $S_{RQ} = \left[\begin{array}{cccc} \mathbf{s}_1 & \mathbf{s}_2 & \cdots & \mathbf{s}_n \end{array}\right]$ can be computed recursively : $\mathbf{s}_1 = \left[\begin{array}{cccc} \frac{\alpha_0}{\beta_0} & 0 & \cdots & 0 \end{array}\right]^T$ and*

$$\mathbf{s}_{k+1} = \alpha_k \cdot C_R(\Theta) \cdot \mathbf{s}_k - a_{k-1,k} \cdot \mathbf{s}_{k-1} - a_{k-2,k} \cdot \mathbf{s}_{k-2} - ... - a_{0,k} \cdot \mathbf{s}_1, \tag{4.6}$$

where $C_R(\Theta)$ is the confederate matrix of an arbitrary polynomial $\Theta(x)$ with respect to R.

COROLLARY 4.3 *Let all the notations in Corollary 4.2 hold, with R now being the power basis, $P = \{1, x, x^2, ..., x^{n-1}\}$. For this simplest case we have $s_{11} = \alpha_0$, and*

$$\mathbf{s}_{k+1} = \alpha_k \cdot Z_0 \cdot \mathbf{s}_k - a_{k-1,k} \cdot \mathbf{s}_{k-1} - ... - a_{0,k} \cdot \mathbf{s}_1, \tag{4.7}$$

where Z_0 is the lower shift matrix. Also in this simplest case the first row of the matrix S_{PQ} is given by

$$\left[\begin{array}{c} s_{1,k} \end{array}\right]_{1 \leq k \leq n} = \left[\begin{array}{cccc} \alpha_0 & \alpha_0\delta_1 & \cdots & \alpha_0\delta_{n-1} \end{array}\right], \tag{4.8}$$

where δ_k are as in (1.10).

Proof. The recurrence (4.6) and its particular case, (4.7) follow easily from (4.5). Comparing (4.7) with (1.11), one obtains (4.8).

□

Note that by definition (4.3), the columns of S_{PQ} are formed from the coefficients of the polynomials $Q_i(x)$ specified by (1.1). Therefore one can recognize (4.7) as simply another form of the recurrence relations (1.1).

4.4. Change of W_Q. Recall that S_{RQ} is the similarity matrix that defines the change of the confederate matrix when passing to another polynomial basis, see, e.g., (4.5). Corollary 4.2 allows us to obtain similar statements for the matrix W_Q in (1.9).

LEMMA 4.4 *Let* $Q = \{Q_0(x), Q_1(x), ..., Q_n(x)\}$ *and* $R = \{R_0(x), R_1(x), ..., R_n(x)\}$ *be systems of polynomials specified by* (1.1) *and* (4.1), *respectively. Then*

$$W_Q = S_{RQ}^{-1} \cdot W_R \cdot S_{RQ}, \tag{4.9}$$

where S_{RQ} *is a similarity matrix, defined as in* (4.2), (4.3). *For the simplest case where* R *stands for the power basis* $P = \{1, x, x^2, ..., x^{n-1}\}$, *we have*

$$W_Q = S_{PQ}^{-1} \cdot Z_0^T \cdot S_{PQ}. \tag{4.10}$$

Proof. Let us first prove (4.10). To this end rewrite (4.7) as

$$s_{22} = \alpha_1 \cdot s_{11}$$

and

$$\begin{bmatrix} s_{1,k} \\ \vdots \\ s_{k-1,k} \\ s_{k,k} \end{bmatrix} = \frac{1}{\alpha_k} \begin{bmatrix} s_{2,k+1} \\ \vdots \\ s_{k,k+1} \\ s_{k+1,k+1} \end{bmatrix} + \frac{a_{k-1,k}}{\alpha_k} \cdot \begin{bmatrix} s_{2,k} \\ \vdots \\ s_{k,k} \\ 0 \end{bmatrix} + ... + \frac{a_{0,k}}{\alpha_k} \cdot \begin{bmatrix} s_{1,1} \\ 0 \\ \vdots \\ 0 \end{bmatrix}.$$

Expressing the last relations in matrix form, we have

$$S_{PQ} \cdot Z_0^T = Z_0^T \cdot S_{PQ} \cdot M_Q \cdot \mathrm{diag}(\frac{1}{\alpha_0}, \frac{1}{\alpha_1}, ..., \frac{1}{\alpha_{n-1}}), \tag{4.11}$$

where M_Q is as in (1.4), which implies (4.10). Furthermore, writing the representation (4.10) for the matrices W_R and W_Q, and then using the obvious identity $S_{RQ} = S_{RP} \cdot S_{PQ}$, one obtains (4.9).

□

Note that equalities (4.8) and (4.10) imply that the general displacement equation in (1.13) can be obtained from its particular case (1.14) by multiplication from the right by the matrix S_{PQ}.

5. INVERSION FORMULAS FOR POLYNOMIAL VANDERMONDE-LIKE MATRICES

5.1. FIRST INVERSION FORMULA

5.1.1. Solution of displacement equation. It is a well known fact that the displacement rank of Toeplitz-like, Cauchy-like and Vandermonde-like matrices is (essentially) inherited under the operation of inversion of a matrix. In this section this result will be extended to polynomial Vandermonde-like matrices. To this end we shall need the following lemma showing how any matrix can be recovered from its $\{D_{\frac{1}{x}}, I, I, W_Q\}$-displacement.

LEMMA 5.1 *Let Q be the system of polynomials specified by* (1.1). *Then for given*

$$G = \begin{bmatrix} g_{ik} \end{bmatrix} \in \mathbf{C}^{n \times \alpha}, B = \begin{bmatrix} b_{ik} \end{bmatrix} \in \mathbf{C}^{\alpha \times n}$$

the unique solution $R \in \mathbf{C}^{n \times n}$ of the equation

$$\nabla_{\{D_{\frac{1}{x}},I,I,W_Q\}}(R) = D_{\frac{1}{x}} \cdot R - R \cdot W_Q = G \cdot B \tag{5.1.1}$$

is given by

$$R = \sum_{i=1}^{\alpha} \mathrm{diag}(c_i) \cdot V_Q(x) \cdot (\sum_{k=1}^{n} d_{i,k} \cdot W_Q^{k-1}), \tag{5.1.2}$$

where

$$c_i = \begin{bmatrix} x_k g_{k,i} \end{bmatrix}_{1 \le k \le n} \in \mathbf{C}^n,$$

and

$$\begin{bmatrix} d_{i,k} \end{bmatrix}_{1 \le k \le n} = \begin{bmatrix} b_{i,k} \end{bmatrix}_{1 \le k \le n} \cdot S_{QP},$$

where $P = \{1, x, x^2, ..., x^n\}$, and S_{QP} is defined in (4.2), (4.3).

Proof. First note that since the spectra of the matrices $D_{\frac{1}{x}}$ and W_Q have no intersection, there is only one solution of equation (5.1.1), see for example [LT, page 411].

Substituting R given by (5.1.2) into (5.1.1) and then using (1.13), we have

$$\nabla_{\{D_{\frac{1}{x}},I,I,W_Q\}}(R) = \sum_{i=1}^{\alpha} \mathrm{diag}(c_i) \cdot (D_{\frac{1}{x}} \cdot V_Q(x) - V_Q(x) \cdot W_Q) \cdot (\sum_{k=1}^{n} d_{i,k} \cdot W_Q^{k-1}) =$$

$$\sum_{i=1}^{\alpha} \mathrm{diag}(c_i) \cdot \begin{bmatrix} \frac{1}{x_1} \\ \frac{1}{x_2} \\ \vdots \\ \frac{1}{x_n} \end{bmatrix} \cdot \begin{bmatrix} \alpha_0 & \alpha_0\delta_1 & \cdots & \alpha_0\delta_{n-1} \end{bmatrix} \cdot (\sum_{k=1}^{n} d_{i,k} \cdot W_Q^{k-1}).$$

Furthermore, using (4.10) and the fact that the first row of S_{PQ} is given by (4.8), we have

$$\nabla_{\{D_{\frac{1}{x}},I,I,W_Q\}}(R) = \sum_{i=1}^{\alpha} \mathrm{diag}(c_i) \cdot \begin{bmatrix} \frac{1}{x_1} \\ \frac{1}{x_2} \\ \vdots \\ \frac{1}{x_n} \end{bmatrix} \cdot \begin{bmatrix} \alpha_0 & \alpha_0\delta_1 & \cdots & \alpha_0\delta_{n-1} \end{bmatrix} \cdot S_{PQ}^{-1} \cdot (\sum_{k=1}^{n} d_{i,k} \cdot (Z_0^T)^{k-1}) \cdot S_{PQ} =$$

$$\sum_{i=1}^{\alpha} \begin{bmatrix} g_{1,i} \\ g_{2,i} \\ \vdots \\ g_{n,i} \end{bmatrix} \cdot \begin{bmatrix} 1 & 0 & \cdots & 0 \end{bmatrix} \cdot (\sum_{k=1}^{n} d_{i,k} \cdot (Z_0^T)^{k-1}) \cdot S_{PQ} = \sum_{i=1}^{\alpha} \begin{bmatrix} g_{1,i} \\ g_{2,i} \\ \vdots \\ g_{n,i} \end{bmatrix} \cdot \begin{bmatrix} d_{i,1} & d_{i,2} & \cdots & d_{i,n} \end{bmatrix} \cdot S_{PQ} =$$

$$\sum_{i=1}^{\alpha} \begin{bmatrix} g_{1,i} \\ g_{2,i} \\ \vdots \\ g_{n,i} \end{bmatrix} \cdot \begin{bmatrix} b_{i,1} & b_{i,2} & \cdots & b_{i,n} \end{bmatrix} = G \cdot B,$$

and (5.1.2) follows.

□

5.1.2. Inversion formula. In the next lemma we show that similar to the situation with other basic classes of structured matrices, the polynomial Vandermonde-like displacement structure is (essentially) preserved under inversion. This fact, and Lemma 4.1 allow us to derive an explicit formula for the inverse of a polynomial Vandermonde-like matrix.

THEOREM 5.2 *Let Q be a system of polynomials specified by* (1.1) *and let* $\nabla_{\{D_{\frac{1}{x}},I,I,W_Q\}}(\cdot)$: $\mathbf{C}^{n\times n} \to \mathbf{C}^{n\times n}$ *be the displacement operator in* (1.13). *Then*

$$\mathrm{rank}\nabla_{\{D_{\frac{1}{x}},I,I,W_Q\}}(R) = \mathrm{rank}\nabla_{\{D_{\frac{1}{x}},I,I,W_{\hat{Q}}\}}(R^{-T}\cdot\tilde{I}), \tag{5.1.3}$$

where $\hat{Q}$ is the associated system specified by (3.1), (3.2). *Moreover, if R is specified by its generator $\{G, B\}$ in the right hand side of*

$$\nabla_{\{D_{\frac{1}{x}},I,I,W_Q\}}(R) = D_{\frac{1}{x}}\cdot R - R\cdot W_Q = G\cdot B, \tag{5.1.4}$$

then

$$R^{-1} = \tilde{I}\cdot\sum_{i=1}^{\alpha}\left(\sum_{k=1}^{n} d_{ik}(W_{\hat{Q}}^T)^{k-1}\right)\cdot V_{\hat{Q}}^T\cdot\mathrm{diag}(c_i), \tag{5.1.5}$$

where c_i and d_ik are determined from 2α linear systems of equations

$$\begin{bmatrix} c_1 & c_2 & \cdots & c_\alpha \end{bmatrix} = D_x\cdot R^{-T}\cdot B^T \in \mathbf{C}^{n\times\alpha}, \qquad \begin{bmatrix} d_{ik} \end{bmatrix} = G^T\cdot R^{-T}\cdot\tilde{I}\cdot S_{\hat{Q}P} \in \mathbf{C}^{\alpha\times n}, \tag{5.1.6}$$

with $D_x = (x_1, x_2, ..., x_n)$.

Proof. Multiplying (5.1.4) by $\tilde{I}\cdot R^{-1}$ from the left and by R^{-1} from the right, and then taking transposes, we obtain

$$D_{\frac{1}{x}}\cdot(R^{-T}\cdot\tilde{I}) - (R^{-T}\cdot\tilde{I})\cdot W_{\hat{Q}} = (R^{-T}B^T)\cdot(G^TR^{-T}\tilde{I}),$$

where we used (3.4). The latter equality means that the $\{D_{\frac{1}{x}}, I, I, W_{\hat{Q}}\}$-displacement rank of the matrix $R^{-T}\cdot\tilde{I}$ is equal to the $\{D_{\frac{1}{x}}, I, I, W_Q\}$-displacement rank of the matrix R, which proves (5.1.3). Writing then formula (5.1.2) for the matrix $R^{-T}\cdot\tilde{I}$, one easily obtains (5.1.5).

□

5.2. SECOND INVERSION FORMULA

5.2.1. Solution of displacement equation. In the previous subsection a formula for the inverse of a polynomial Vandermonde-like matrix was obtained using its $\{D_{\frac{1}{x}}, I, I, W_Q\}$-displacement. Similarly, we shall obtain below another formula for the inverse of polynomial Vandermonde-like matrix, but now using its $\{D_x, I, I, C_Q(Q_n)\}$-displacement representation.

As in the previous subsection, the derivation of an inversion formula is based on an expansion for a polynomial Vandermonde-like matrix from its generator. For this, we shall need the next auxiliary lemma, which is an extension of a proposition in [DFZ].

LEMMA 5.3 *Let Q be a polynomial system specified by* (1.1), *and $C_Q(Q_n)$ be the confederate matrix as in* (1.6) *of $Q_n(x)$ with respect to Q. Then the first column of the matrix $Q_k(C_Q(Q_n))$ is given by*

$$Q_k(C_Q(Q_n))\cdot e_1 = \alpha_0\cdot e_{k+1}. \tag{5.2.1}$$

Let $\hat{Q}$ be the system of associated polynomials given by (3.1), (3.2)*. Then the last row of the matrix $\hat{Q}_k(C_Q(Q_n))$ is given by*

$$e_n^T \cdot \hat{Q}_k(C_Q(Q_n)) = \alpha_n \cdot e_{n-k}^T. \tag{5.2.2}$$

Proof. Observe that since the matrix $C_Q(Q_n))$ is of an upper Hessenberg form, the last $n-k-1$ entries of all the vectors $C_Q(Q_n)) \cdot e_1$, $C_Q(Q_n))^2 \cdot e_1$, ... , $C_Q(Q_n))^k \cdot e_1$ are zero. Clearly the latter statement holds also for the vector $Q_k(C_Q(Q_n))) \cdot e_1$.

Furthermore, let us recall that, in accordance with [MB], polynomial $Q_k(x)$ is the characteristic polynomial of the upper left submatrix $A_k \in \mathbf{C}^{k\times k}$ of $C_Q(Q_n))$ for $k = 1, 2, ..., n$. Therefore by the Cayley-Hamilton theorem, the polynomial $Q_k(x)$ annihilates the matrix A_k. It is easy to check that because of the Hessenberg structure (1.6) of the matrix $C_Q(Q_n))$, the first k components of the vectors $A_k^j \cdot e_1$ and $C_Q(Q_n))^j \cdot e_1$ are the same for $0 \le j \le k$. Hence the first k components of the vectors $Q_k(A_k) \cdot e_1$ and $A_n^J \cdot e_1 = Q_k(C_Q(Q_n))) \cdot e_1$ also coincide, i.e. are all zero. Finally, the fact that the only nonzero $(k+1)$-th entry of $Q_k(C_Q(Q_n))) \cdot e_1$ is equal to α_0 is easily deduced using induction, and (5.2.1) follows. Furthermore, the equality (5.2.2) immediately follows from (5.2.1) and assertions (i), (ii) of Lemma 3.1.

□

In the next lemma we show how any matrix can be recovered from its $\{D_x, I, I, C_Q(Q_n)\}$-generator.

LEMMA 5.4 *Let Q and the systems of polynomials, specified by* (1.1) *and $C_Q(Q_n)$ be the confederate matrix of $Q_n(x)$ with respect to Q. Let*

$$G = \begin{bmatrix} g_{ik} \end{bmatrix} \in \mathbf{C}^{n\times\alpha}, \qquad B = \begin{bmatrix} b_{ik} \end{bmatrix} \in \mathbf{C}^{\alpha\times n}.$$

If $Q_n(x_k) \neq 0$ for $k = 1, 2, ...n$, then the unique solution of the equation

$$\nabla_{\{D_x,I,I,C_Q(Q_n)\}}(R) = D_x \cdot R - R \cdot C_Q(Q_n) = G \cdot B \tag{5.2.3}$$

is given by

$$R = \sum_{k=1}^{\alpha} \mathrm{diag}(\frac{g_{1,k}}{Q_n(x_1)}, \frac{g_{2,k}}{Q_n(x_2)}, ..., \frac{g_{n,k}}{Q_n(x_n)}) \cdot V_Q(x) \cdot (\sum_{j=1}^{n} b_{k,n-j+1}\hat{Q}_{j-1}(C_Q(Q_n))). \tag{5.2.4}$$

where $\hat{Q} = \{\hat{Q}_0(x), \hat{Q}_1(x), ..., \hat{Q}_n(x)\}$ is the associated system of polynomials, given by (3.1), (3.2).

Proof. First note that the spectrum of the confederate matrix $C_Q(Q_n)$ coincides with the zeros of the polynomial $Q_n(x)$ and therefore has no intersection with the spectrum of the matrix D_x. Hence there is only one solution of the equation (5.2.3), see for example [LT, page 411].

Substituting R given by (5.2.4) into (5.2.3), we have

$$\nabla_{\{D_x,I,I,C_Q(Q_n)\}}(R) = \sum_{k=1}^{\alpha} \mathrm{diag}(\frac{g_{1,k}}{Q_n(x_1)}, \frac{g_{2,k}}{Q_n(x_2)}, ..., \frac{g_{n,k}}{Q_n(x_n)}) \cdot$$

$$(D_x \cdot V_Q(x) - V_Q(x) \cdot C_Q(Q_n(x))) \quad \cdot \quad (\sum_{j=1}^{n} b_{k,n-j+1}\hat{Q}_{j-1}(C_Q(Q_n))),$$

where we have used the fact that the matrix $C_Q(Q_n)$ commutes with the polynomial $\hat{Q}_{j-1}(C_Q(Q_n))$ in $C_Q(Q_n)$. Furthermore, applying (1.7), we have

$$\nabla_{\{D_x,I,I,C_Q(Q_n)\}}(R) = \sum_{k=1}^{\alpha} \mathrm{diag}(\frac{g_{1,k}}{Q_n(x_1)}, \frac{g_{2,k}}{Q_n(x_2)}, ..., \frac{g_{n,k}}{Q_n(x_n)}) \cdot \begin{bmatrix} Q_n(x_1) \\ Q_n(x_2) \\ \vdots \\ Q_n(x_n) \end{bmatrix}.$$

$$\begin{bmatrix} 0 & 0 & \cdots & \frac{1}{\alpha_n} \end{bmatrix} \cdot (\sum_{j=1}^{n} b_{k,n-j+1} \hat{Q}_{j-1}(C_Q(Q_n))) = \sum_{k=1}^{\alpha} \begin{bmatrix} g_{1,k} \\ g_{2,k} \\ \vdots \\ g_{n,k} \end{bmatrix} \cdot \begin{bmatrix} b_{k,1} & b_{k,2} & \cdots & b_{k,n} \end{bmatrix} = G \cdot B.$$

where we have used (5.2.2).

□

5.2.2. Inversion formula. Lemma 5.4 allows us to write down another inversion formula for polynomial Vandermonde-like matrix.

THEOREM 5.5 *Let Q be the polynomial system specified by* (1.1), *and* $\nabla_{\{D_x,I,I,C_Q(Q_n)\}}(\cdot) : \mathbf{C}^{n\times n} \to \mathbf{C}^{n\times n}$ *be the displacement operator in* (1.7). *Then*

$$\mathrm{rank}\nabla_{\{D_x,I,I,C_Q(Q_n)\}}(R) = \mathrm{rank}\nabla_{\{D_x,I,I,C_{\hat{Q}}(Q_n)\}}(R^{-T} \cdot \tilde{I}), \tag{5.2.5}$$

where $\hat{Q}$ is the associated polynomial system, given by (3.1). (3.2). *Moreover, if R is specified by its generator $\{G, B\}$ on the right hand side of*

$$\nabla_{\{D_x,I,I,C_Q(Q_n)\}}(R) = D_x \cdot R - R \cdot C_Q(Q_n) = G \cdot B, \tag{5.2.6}$$

then

$$R^{-1} = \tilde{I} \cdot \sum_{k=1}^{\alpha}(\sum_{j=1}^{n} d_{k,j} \cdot Q_{j-1}(C_{\hat{Q}}^T(Q_n))) \quad \cdot \quad V_{\hat{Q}}^T \cdot \mathrm{diag}(c_k), \tag{5.2.7}$$

where $c_i \in \mathbf{C}^{n\times 1}$ and $d_{ki} \in \mathbf{C}$ are determined from 2α linear systems of equations

$$\begin{bmatrix} c_1 & c_2 & \cdots & c_\alpha \end{bmatrix} = \mathrm{diag}(\frac{1}{Q_n(x_1)}, \frac{1}{Q_n(x_2)}, ..., \frac{1}{Q_n(x_n)}) \cdot R^{-T} \cdot B^T \in \mathbf{C}^{n\times\alpha}, \tag{5.2.8}$$

$$\begin{bmatrix} d_{k,i} \end{bmatrix} = G^T \cdot R^{-T} \in \mathbf{C}^{\alpha\times n}. \tag{5.2.9}$$

Proof. Multiplying (5.2.6) by $\tilde{I} \cdot R^{-1}$ from the left and by R^{-1} from the right and then taking transposes, we obtain

$$D_x \cdot R^{-T} - R^{-T} \cdot C_{\hat{Q}}(Q_n) = (R^{-T} B^T) \cdot (G^T R^{-T} \tilde{I}),$$

where we used (3.3). The latter equality means that the $\{D_x, I, I, C_{\hat{Q}}(Q_n)\}$-displacement rank of the matrix $R^{-T} \cdot \tilde{I}$ is equal to the $\{D_x, I, I, C_Q(Q_n(x))\}$-displacement rank of the matrix R, which proves (5.2.5).

Furthermore, by writing formula (5.2.4) for the matrix $R^{-T} \cdot \tilde{I}$, one easily obtains (5.2.7).

□

6. FAST GEPP FOR POLYNOMIAL VANDERMONDE-LIKE MATRICES

6.1. Generator recursion. The standard Gaussian elimination procedure applied to an arbitrary matrix R_1 is based on executing $n-1$ recursive steps as in

$$R_1 = \begin{bmatrix} d_1 & u_1 \\ l_1 & R_{22}^{(1)} \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ \frac{1}{d_1} l_1 & I \end{bmatrix} \cdot \begin{bmatrix} d_1 & u_1 \\ 0 & R_2 \end{bmatrix}, \tag{6.1}$$

where $R_2 = R_{22}^{(1)} - \frac{1}{d_1} l_l u_1$ is the Schur complement of nonzero (1,1) entry d_1 in the matrix R_1. The step (6.1) gives the first column $\begin{bmatrix} 1 \\ \frac{1}{d_1} l_1 \end{bmatrix}$ of L and the first row $\begin{bmatrix} d_1 & u_1 \end{bmatrix}$ of U in the LU factorization of R_1. Proceeding with the Schur complement R_2 recursively, one can compute the whole LU decomposition for R_1 in $O(n^3)$ operations.

It is well known that when R_1 possesses a displacement structure, the Gaussian elimination procedure can be speeded up, requiring only $O(n^2)$ operations. The remarkable fact that makes this speed up possible is that the displacement structure of a matrix is inherited by successive Schur complements. This result can be traced back to the Schur work [S], where it was derived for Toeplitz matrices (however in a rather implicit form; the concept of displacement was introduced in [KKM] 60 years later). Explicitly this result was obtained by Morf in [M] for the Toeplitz choice $\nabla_{\{I,I,Z,Z^T\}}$ for the displacement operator in (1.2), and later it was further generalized by different authors. In particular it was extended in [CK] to the Sylvester form (1.3) of displacement operator with lower triangular Ω and upper triangular A. In the most general form this statement can be found in [KS1], where it appeared for the generalized displacement structure of the form (1.2) with lower triangular Ω, F and upper triangular Δ, A. Moreover a generalized Schur algorithm from [KS1] provides this result with a constructive proof. The next lemma, from [GO4], [GKO], presents a variant of the generalized Schur algorithm, corresponding to the Sylvester form of displacement operator.

LEMMA 6.1 *Let matrix* $R_1 = \begin{bmatrix} d_1 & u_1 \\ l_1 & R_{22}^{(1)} \end{bmatrix}$ *satisfies the Sylvester type displacement equation*

$$\Omega_1 \cdot R_1 - R_1 \cdot A_1 = G_1 \cdot B_1, \qquad (G_1 \in \mathbf{C}^{n \times \alpha}, \quad B_1 \in \mathbf{C}^{\alpha \times n}). \tag{6.2}$$

If (1,1) *entry* d_1 *of* R_1 *is nonzero, then the Schur complement* $R_2 = R_{22}^{(1)} - \frac{1}{d_1} l_1 u_1$ *satisfies the Sylvester type displacement equation*

$$\Omega_2 \cdot R_2 - R_2 \cdot A_2 = G_2 \cdot B_2, \tag{6.3}$$

where Ω_2 *and* A_2 *are obtained from* $\Omega_1 = \begin{bmatrix} \omega_1 & 0 \\ * & \Omega_2 \end{bmatrix}$, $A_1 = \begin{bmatrix} a_1 & * \\ 0 & A_2 \end{bmatrix}$, *by deleting the first row and column, and*

$$\begin{bmatrix} 0 \\ G_2 \end{bmatrix} = G_1 - \begin{bmatrix} 1 \\ \frac{1}{d_1} l_1 \end{bmatrix} \cdot g_1, \qquad \begin{bmatrix} 0 & B_2 \end{bmatrix} = B_1 - b_1 \cdot \begin{bmatrix} 1 & \frac{1}{d_1} u_1 \end{bmatrix}, \tag{6.4}$$

where g_1 *and* b_1 *are the first row of* G_1 *and the first column of* B_1, *respectively.*

The pair of matrices $\{G_1, B_1\}$ on the right hand side of (6.2) is called a *generator* of R_1. Formulas (6.4) allow an implementation of the step (6.1) of the Gaussian elimination procedure, in which computing $(n-1)^2$ entries of a Schur complement R_2 is replaced by computing $2\alpha(n-1)$ entries of its generator $\{G_2, B_2\}$.

In the rest of this section we shall specify such an implementation for a polynomial Vandermonde-like matrix R_1 satisfying the displacement equation (6.2) (see also (1.13)), in which

$$\Omega_1 = D_{\frac{1}{x}}, \qquad A_1 = W_Q = N_Q \cdot M_Q^{-1}, \tag{6.5}$$

where W_Q, $D_{\frac{1}{x}}$, N_Q and M_Q are given by (1.9), (1.10) and (1.4). In order to design an algorithm one has to answer the next two questions.

- First, how to recover the first row and column of R_1 from its generator (this allows us write down the first row and column in the LU decomposition (6.1), and to run the generator recursion (6.4)).
- Second, how to obtain truncated matrices Ω_2 and A_2 from Ω_1 and A_1 given by (6.5) (recall that in our case Ω_1, A_1 are not given explicitly).

To answer the first question, let us multiply by e_1 from the right the equation (6.2) with Ω_1 and A_1 as in (6.5). Then one obtains the following expression for the first column of R_1

$$\begin{bmatrix} d_1 \\ l_1 \end{bmatrix} = D_x \cdot G_1 \cdot b_1, \tag{6.6}$$

where $D_x = \mathrm{diag}(x_1, ..., x_n)$ and $b_1 \in C^{1\times\alpha}$ is the first column of B_1. Similarly multiplying by e_1^T from the left the equation (6.2) in which Ω_1 and A_1 as in (6.5), one obtains

$$\begin{bmatrix} d_1 & u_1 \end{bmatrix} \cdot \left(\frac{1}{x_1} \cdot M_Q - N_Q\right) = g_1 \cdot B_1 \cdot M_Q, \tag{6.7}$$

where g_1 is the first row of G_1.

Finally, by the upper triangular form of M_Q, we have

$$\Omega_2 = \mathrm{diag}(x_2, ..., x_n), \qquad A_2 = N_{Q,2} \cdot M_{Q,2}^{-1}, \tag{6.8}$$

where $N_{Q,2}$ and $M_{Q,2}$ are obtained by deleting the first row and column from N_Q and M_Q, respectively. Using these arguments we shall write down an implementation of Gaussian elimination for polynomial Vandermonde-like matrices in subsection 6.3. However Gaussian elimination is in general an unstable algorithm and to stabilize it, various pivoting techniques are applied. In the next subsection we shall show how partial pivoting can be incorporated into the implementation of Sec. 6.1.

6.2. Partial pivoting. The standard way to cope with error accumulation in Gaussian elimination is to apply the partial pivoting technique, i.e. to maximize the (1,1) entry of a matrix by means of row permutations, and then to repeat this procedure for each of the successive Schur complements. Of course a row permutation can destroy the structure of a matrix, as is certainly true for Toeplitz–like matrices. In a recent paper [GKO] it was observed that partial pivoting can be easily incorporated into implementations of Gaussian elimination that exploit a displacement structure of the form

$$\Omega_1 \cdot R_1 - R_1 \cdot A_1 = G_1 \cdot B_1, \tag{6.9}$$

where Ω_1 is a diagonal matrix : $\Omega_1 = \mathrm{diag}(t_1, t_2, ..., t_n)$. Indeed, interchange of the 1-st and k-th rows of R_1 is equivalent to multiplication by a corresponding permutation matrix P. It is easy to see that after a row permutation, the new matrix $P \cdot R_1$ satisfies the displacement equation (6.9) with the diagonal matrix Ω_1 replaced by the diagonal matrix $P \cdot \Omega_1 \cdot P^T$ and with G_1 replaced by $P \cdot G_1$. This means that a row interchange does not destroy the displacement structure of polynomial Vandermonde-like matrix. In fact it allows us to incorporate partial pivoting into above implementation of Gaussian elimination.

6.3. GEPP for polynomial Vandermonde-like matrix. Let a polynomial Vandermonde-like matrix R_1 be specified by matrices Ω_1 and A_1 as in (6.5) and by its generator $\{G_1, B_1$ on the right hand side of displacement equation (6.2). Then the arguments in subsections 5.1, 5.2 lead to the following algorithm.

Implementation of step (6.1) of the GEPP procedure

1). Compute the first column of R_1 via (6.6).

2). Find, say in (k,1) position, the maximum magnitude entry in the computed first column of R_1.
Swap the 1-st and the k-the entries of the first column.
Swap the (1,1) and (k,k) entries of Ω_1.
Swap the 1-st and k-th rows of G_1.

3). Compute the first row of R_1 by solving the triangular linear system (6.7) by back-substitution.

4). Write down the first column $\begin{bmatrix} 1 \\ \frac{1}{d_1} l_1 \end{bmatrix}$ of L and first row $\begin{bmatrix} d_1 & u_1 \end{bmatrix}$ of U in the LU decomposition of R_1.
Let P_1 be the permutation matrix of the 1-st and k-th entries.

5). Compute a generator $\{G_2, B_2\}$ for the Schur complement R_2 by using (6.4).

Proceeding recursively with arrays $\{G_2,\ B_2\}$, and Ω_2, A_2 as in (6.8), one finally computes the factorization $R_1 = P \cdot L \cdot U$, where $P = P_1 \cdot P_2 \cdot ... \cdot P_{n-1}$, and P_k is a permutation of the k-th step of the recursion. Then a linear system with R_1 can be solved in $O(n^2)$ operations by forward and back-substitution. We remark that the proposed algorithm is not restricted to strongly regular matrices and is valid for an arbitrary invertible matrix.

6.4. Complexity. Analysis of the above scheme shows that its complexity depends on the complexity of the matrix M_Q, which in turn depends upon the length of the recurrence relations (1.1) of the system of polynomials Q. When these polynomials satisfy m-term recurrence relations, the overall complexity of the algorithm is $O(m\alpha n^2)$ arithmetic operations.

7. GENERALIZED PARKER-FORNEY-TRAUB ALGORITHM FOR INVERSION OF POLYNOMIAL VANDERMONDE-LIKE MATRICES

As was mentioned in the introduction, if an ordinary Vandermonde matrix $V_P(x)$ is invertible, then all n^2 entries of its inverse can be computed in $O(n^2)$ arithmetic operations by a now well-known algorithm independently discovered by many authors [P], [F], [Tr] (see also [GO5]). Later this result was extended to Chebyshev–Vandermonde matrices in [GO2] and to three-term Vandermonde matrices in [CR]. In this section this result will be extended to *m-term Vandermonde matrices*, i.e. to matrices of the form (0.1) with Q satisfying recurrence relations (1.1), in which only the first m coefficients $a_{k-1,k}, a_{k-2,k}, ..., a_{0,k}$ may differ from zero. Moreover we shall further extend this result to *m-term Vandermonde-like matrices*, showing that the complexity of inverting of such a matrix is $O(m\alpha n^2)$ operations, where α is the displacement rank of the matrix.

THEOREM 7.1 *Let Q be a system of polynomials satisfying m-term recurrence relations, and let matrices W_Q and $D_{\frac{1}{x}}$ be given by (1.9) and (1.12), respectively. Let the m-term Vandermonde-like matrix R be given by its $\{D_{\frac{1}{x}}, I, I, W_Q\}$-generator $G \in \mathbf{C}^{n\times\alpha}$, $B \in \mathbf{C}^{\alpha\times n}$ on the right hand side of*

$$\nabla_{\{D_{\frac{1}{x}}, I, I, W_Q\}}(R) = D_{\frac{1}{x}} \cdot R - R \cdot W_Q = G \cdot B.$$

Then all n^2 entries of R^{-1} can be computed in $O(\alpha m n^2)$ operations.

Proof. The proof of the theorem is based on the representation (5.1.5) of R^{-1}. In order to compute the parameters $c_i \in \mathbf{C}^n$ and $d_{ik} \in \mathbf{C}$ in (5.1.5), one has to solve 2α linear systems in (5.1.6) involving matrices R and $S_{P\hat{Q}}$. The entries of upper triangular matrix $S_{P\hat{Q}}$ can be computed by (4.7) in mn^2 operations, then the linear system with $S_{P\hat{Q}}$ can be solved $O(n^2)$ in operations via back-substitutionin. Furthermore, a triangular factorization of R can be computed in $O(\alpha m n^2)$ operations via the implementation of GEPP from section 6.3. Then a linear system with R can be solved in $O(n^2)$ operations. Thus the overall cost of computing the parameters $c_i \in \mathbf{C}^n$ and $d_{ik} \in \mathbf{C}$ in (5.1.5) is no more than $O(m\alpha n^2)$ operations.

It remains to compute the expression on the right hand side of (5.1.5). Multiplication of a matrix by a diagonal matrix as well as adding matrices are cheap operations and require only $O(n^2)$ operations. The main computational burden in (5.1.5) is computing the product of a polynomial in $W_{\hat{Q}}^T$ by the matrix $V_{\hat{Q}}(x)^T$. In the next lemma we show that this complexity is no more than $O(mn^2)$ operations, which will prove the Theorem.

□

LEMMA 7.2 *Let a system of polynomials Q satisfy the m-term recurrence relations, the matrices $V_Q(x)$ and W_Q be defined as in (0.1) and (1.9), and let numbers $d_1, d_2, ..., d_n$ be arbitrary. Then the complexity of computing the entries of the matrix $V_Q(x) \cdot (\sum_{k=1}^n d_k \cdot W_Q^{k-1})$ is no more than $O(mn^2)$ operations.*

Proof. Let us observe that in accordance with (1.13) the matrix $V_Q(x) \cdot W_Q$ is a rank-one perturbation of a low complexity matrix $D_{\frac{1}{x}} \cdot V_Q(x)$, i.e.,

$$V_Q(x) \cdot W_Q = D_{\frac{1}{x}} \cdot V_Q(x) - \begin{bmatrix} \frac{1}{x_1} \\ \frac{1}{x_2} \\ \vdots \\ \frac{1}{x_n} \end{bmatrix} \cdot \begin{bmatrix} s_{11} & s_{12} & \cdots & s_{1n} \end{bmatrix}, \tag{7.1}$$

where we used the fact that the first row of the matrix $S_{PQ} = \begin{bmatrix} s_{ij} \end{bmatrix}_{1\le i,j\le n}$ has the form (4.8). Using (7.1) we further have

$$V_Q(x) \cdot W_Q^2 = D_{\frac{1}{x}} \cdot V_Q(x) \cdot W_Q - \begin{bmatrix} \frac{1}{x_1} \\ \frac{1}{x_2} \\ \vdots \\ \frac{1}{x_n} \end{bmatrix} \cdot \begin{bmatrix} s_{11} & s_{12} & \cdots & s_{1n} \end{bmatrix} \cdot W_Q =$$

$$= D^2_{\frac{1}{x}} \cdot V_Q(x) - \begin{bmatrix} \frac{1}{x_1^2} \\ \frac{1}{x_2^2} \\ \vdots \\ \frac{1}{x_n^2} \end{bmatrix} \cdot \begin{bmatrix} s_{11} & s_{12} & \cdots & s_{1n} \end{bmatrix} - \begin{bmatrix} \frac{1}{x_1} \\ \frac{1}{x_2} \\ \vdots \\ \frac{1}{x_n} \end{bmatrix} \cdot \begin{bmatrix} s_{21} & s_{22} & \cdots & s_{2n} \end{bmatrix}, \tag{7.2}$$

where we used the equality $\begin{bmatrix} s_{11} & s_{12} & \cdots & s_{1n} \end{bmatrix} \cdot W_Q = \begin{bmatrix} s_{21} & s_{22} & \cdots & s_{2n} \end{bmatrix}$, which easily follows from (4.10). Proceeding similarly, one obtains

$$V_Q(x) \cdot W_Q^k = D^k_{\frac{1}{x}} \cdot V_Q(x) - \begin{bmatrix} \frac{1}{x_1^k} \\ \frac{1}{x_2^k} \\ \vdots \\ \frac{1}{x_n^k} \end{bmatrix} \cdot \begin{bmatrix} s_{11} & s_{12} & \cdots & s_{1n} \end{bmatrix} -$$

$$\begin{bmatrix} \frac{1}{x_1^{k-1}} \\ \frac{1}{x_2^{k-1}} \\ \vdots \\ \frac{1}{x_n^{k-1}} \end{bmatrix} \cdot \begin{bmatrix} s_{21} & s_{22} & \cdots & s_{2n} \end{bmatrix} - \ldots - \begin{bmatrix} \frac{1}{x_1} \\ \frac{1}{x_2} \\ \vdots \\ \frac{1}{x_n} \end{bmatrix} \cdot \begin{bmatrix} s_{k1} & s_{k2} & \cdots & s_{kn} \end{bmatrix}.$$

From the latter equality it follows that

$$V_Q(x) \cdot \sum_{k=1}^{n} d_k \cdot W_Q^{k-1} = (\sum_{k=1}^{n} d_k \cdot D^{k-1}_{\frac{1}{x}}) \cdot V_Q(x) - V_F(\frac{1}{x}) \cdot S_{PQ}, \tag{7.3}$$

where

$$V_F(\frac{1}{x}) = \begin{bmatrix} F_0(\frac{1}{x_1}) & F_1(\frac{1}{x_1}) & \cdots & F_{n-1}(\frac{1}{x_1}) \\ F_0(\frac{1}{x_2}) & F_(\frac{1}{x_2}) & \cdots & F_{n-1}(\frac{1}{x_2}) \\ \vdots & \vdots & & \vdots \\ F_0(\frac{1}{x_n}) & F_1(\frac{1}{x_n}) & \cdots & F_{n-1}(\frac{1}{x_n}) \end{bmatrix},$$

with $F_{n-1}(\frac{1}{x}) = 0$ and

$$F_k(\frac{1}{x}) = \frac{1}{x} \cdot (F_{k+1}(\frac{1}{x}) + d_{k+2}) \qquad (k = 1, 2, ..., n-1). \tag{7.4}$$

We also set $F_{-1}(\frac{1}{x}) = F_0(\frac{1}{x}) + d_1$. The recurrence relations (7.4) allow to compute the entries of the matrix $V_F(\frac{1}{x})$ in $2n^2$ operations. A further n operations give us the entries of the diagonal matrix $\sum_{k=1}^{n} d_k \cdot D^{k-1}_{\frac{1}{x}} = \mathrm{diag}(F_{-1}(\frac{1}{x_1}), F_{-1}(\frac{1}{x_2}), ..., F_{-1}(\frac{1}{x_n 2}))$. Multiplication of the latter diagonal matrix by $V_Q(x)$ is performed in n^2 operations, which gives us the first term in the right hand side of (7.3). What remains is to show that the second term in the right hand side of (7.3), i.e. the product

$$H = \begin{bmatrix} h_{ij} \end{bmatrix} = V_F(\frac{1}{x}) \cdot S_{PQ} \tag{7.5}$$

can be computed in $O(n^2)$ operations. To this end first observe that the entries of the matrix $S_{PQ} = \begin{bmatrix} s_{ij} \end{bmatrix}$ can be computed via (4.7) in $O(mn^2)$ operations. The complexity $O(n^2)$ is needed for computing the auxiliary quantities

$$\begin{bmatrix} c_1 & c_2 & \cdots & c_n \end{bmatrix} = \begin{bmatrix} d_2 & d_3 & \cdots & d_n & 0 \end{bmatrix} \cdot S_{PQ},$$

which we shall need in what follows. Having the entries of S_{PQ} and the numbers $c_1, c_2, ..., c_n$, the entries of the i-th row of H in (7.5) are computed as follows. Clearly,

$$h_{i,1} = F_0(\frac{1}{x_i}) \cdot s_{11}, \qquad h_{i,2} = F_0(\frac{1}{x_i}) \cdot s_{12} + F_1(\frac{1}{x_i}) \cdot s_{22}, \tag{7.6}$$

and from (7.5), (4.7) it follows that

$$h_{i,k+1} = \alpha_k \cdot (x_i \cdot h_{i,k} - c_k) - a_{k-1,k} \cdot h_{i,k} - ... - a_{0,k} \cdot h_{i,1} \qquad (k = 2, 3, ..., n-1), \tag{7.7}$$

where we used the fact that from (7.4) it follows that

$$\begin{bmatrix} F_0(\frac{1}{x_i}) & F_1(\frac{1}{x_i}) & \cdots & F_{n-1}(\frac{1}{x_i}) \end{bmatrix} \cdot \begin{bmatrix} 0 \\ s_{1k} \\ \vdots \\ s_{n-1,k} \end{bmatrix} = \begin{bmatrix} F_1(\frac{1}{x_i}) & F_2(\frac{1}{x_i}) & \cdots & F_{n-1}(\frac{1}{x_i}) & 0 \end{bmatrix} \cdot \begin{bmatrix} s_{1,k} \\ s_{2,k} \\ \vdots \\ s_{n,k} \end{bmatrix} =$$

$$= (x_i \cdot \begin{bmatrix} F_0(\frac{1}{x_i}) & F_1(\frac{1}{x_i}) & \cdots & F_{n-1}(\frac{1}{x_i}) \end{bmatrix} - \begin{bmatrix} d_2 & d_3 & \cdots & d_n & 0 \end{bmatrix}) \cdot \begin{bmatrix} s_{1,k} \\ s_{2,k} \\ \vdots \\ s_{n,k} \end{bmatrix} = x_i \cdot h_{i,k} - \sum_{j=1}^{n-1} d_{j+1} \cdot s_{j,k} = x \cdot h_{i,k} - c_k.$$

Thus the entries of H are computed by (7.6), (7.7) in $O(mn^2)$ operations, which proves the assertions of the Lemma.

□

8. TRANSFORMATION INTO CAUCHY-LIKE MATRICES

As was mentioned in the introduction, it was observed by Heinig and Rost in [HR] that an ordinary Cauchy matrix $C(x, y) = \begin{bmatrix} \frac{1}{x_i - y_j} \end{bmatrix}$ satisfies the equation

$$\nabla_{\{D_x, I, I, D_y\}}(C(x, y)) = D_x \cdot C(x, y) - C(x, y) \cdot D_y = \begin{bmatrix} 1 \\ 1 \\ \vdots \\ 1 \end{bmatrix} \cdot \begin{bmatrix} 1 & 1 & \cdots & 1 \end{bmatrix},$$

where $D_x = \text{diag}(x_1, x_2, ..., x_n)$ and $D_y = \text{diag}(y_1, y_2, ..., y_n)$. Using this observation, *Cauchy–like matrices* were introduced in [HR] as matrices with low $\{D_x, I, I, D_y\}$-displacement rank. Since then various fast $O(n^2)$ algorithms have been designed for solving linear systems with Cauchy-like coefficient matrices, see e.g. [HR], [GKK], [GKKL], [He], [GO4], [GKO]. In particular, a fast $O(n^2)$ implementation of Gaussian elimination with partial pivoting was designed recently in [GO4], [GKO] for Cauchy-like matrices. Furthermore it was shown that a matrix from any other structured class can be transformed into a Cauchy-like matrix by performing computations on the generator of a matrix. On this basis fast $O(n^2)$ algorithms for solving a linear system were designed for Toeplitz-like, Hankel-like, Toeplitz-plus-Hankel-like and Vandermonde-like matrices in [GKO] and for Chebyshev-Vandermonde-like matrices in [KO1]. In this section we shall derive an analogous result for polynomial Vandermonde-like matrices.

LEMMA 8.1 *Let $Q = \{Q_0(x), ..., Q_{n-1}(x)\}$ be a system of polynomials satisfying* (1.1), *and let $\Theta(x) = (x - z_1)(x - z_2)...(x - z_n)$, where $z_1, z_2, ..., z_n$ are any n pairwise distinct numbers. Let* *matrix R be given by its $\{D_x, I, I, C_Q(\Theta)\}$-generator $G \in \mathbf{C}^{n\times\alpha}$, $B \in \mathbf{C}^{\alpha\times n}$ on the right hand side of*

$$\nabla_{\{D_x,I,I,C_Q(\Theta)\}}(R) = D_x \cdot R - R \cdot C_Q(\Theta) = G \cdot B. \tag{8.1}$$

Then $R \cdot V_Q(z)^{-1}$ is a Cauchy-like matrix :

$$\nabla_{\{D_x,I,I,D_z\}}(R \cdot V_Q(z)^{-1}) = D_x \cdot (R \cdot V_Q(z)^{-1}) - (R \cdot V_Q(z)^{-1}) \cdot D_z = G \cdot (B \cdot V_Q(z)^{-1}), \tag{8.2}$$

where $D_z = \text{diag}(z_1, z_2, ..., z_n)$.

Proof. Recall that matrix $C_Q(\Theta)$ is diagonalized by $V_Q(z)$: $C_Q(\Phi) = V_Q(z)^{-1} \cdot D_z \cdot V_Q(z)$, see, e.g., (1.7). Substituting the latter expression into (8.1) and then multiplying by $V_Q(z)^{-1}$ from the right, one obtains (8.2).

□

Theorem 8.1 is valid for an arbitrary polynomial Vandermonde-like matrix R. However in many special cases the corresponding matrix $V_Q(z)$ is simple in form, so that this theorem suggests one more efficient algorithm for solving a linear system with such R. As was mentioned in the introduction, the matrix $V_Q(z)$ is simple in form for many important families of polynomials Q. For example in the case where the $Q_i(x) = x^i$ and $z_i = exp(\frac{2\pi i}{n})$, $V_Q(z)$ is the (normalized) DFT matrix. Similarly when $Q_i(x)$ are Chebyshev polynomials of the first kind and $z_i = \frac{2\pi i}{n}$, $V_Q(z)$ is the DCT matrix. Finally in the case where the $Q_i(x)$ are Chebyshev polynomials of the second kind and $z_i = \frac{2\pi i}{n}$, $V_Q(z)$ is the DST matrix times a certain diagonal matrix. In all these cases the transformation of the polynomial Vandermonde-like matrix R into a Cauchy-like matrix $R \cdot V_Q(z)^{-1}$ is reduced to computing for the latter Cauchy-like matrix the $\{D_x, I, I, D_z\}$-generator $\{G, (B\dot{V}_Q(z)^{-1})\}$ by applying a certain fast transform (see e.g. [GKO] and [KO1]). Applying to this Cauchy-like matrix the fast GEPP algorithm from [GKO], we compute the factorization

$$R = P \cdot L \cdot U \cdot V_Q(z).$$

Making use of the above factorization one can solve a linear system with coefficient matrix R in $O(n^2)$ operations via forward and back-substitution and then applying a certain fast transform associated with $V_Q(z)$.

ACKNOWLEDGMENT

We would like to thank an anonymous referee for his useful suggestions, allowing us to improve the exposition of the paper, and for the short proof of Lemma 2.3.

REFERENCES

[B] S.Barnett, A companion matrix analogue for orthogonal polynomials, *Linear Algebra Appl.*, **12** : 197 – 208 (1975).

[BP] A. Björck and V.Pereyra, Solution of Vandermonde systems of linear equations, *Math. of Computation*, **24** : 893 – 903 (1970).

[BiP] D. Bini and V.Pan, *Numerical and Algebraic Computations with Matrices and Polynomials*, Birkhauser, Boston, MA, 1994.

[BKO] T.Boros, T.Kailath and V.Olshevsky, Fast algorithms for solving Vandermonde and Chebyshev-Vandermonde linear equations, preprint, 1994.

[CK] J.Chun and T.Kailath, Fast triangularization and orthogonalization of Hankel and Vandermonde matrices, *Linear Algebra Appl.*, **151** : 199 – 228 (1991).

[CR] D.Calvetti and L.Reichel, Fast inversion of Vandermonde-like matrices involving orthogonal polynomials, *BIT*, **33** : 473 – 484 (1993).

[DFZ] C.Di Fiore and P.Zellini, Matrix decompositions using displacement rank and classes of commutative matrix algebras, preprint, 1994.

[F] G.Forney, *Concatenated codes*, The M.I.T. Press, 1966, Cambridge.

[G] W.Gautschi, The condition of Vandermonde–like matrices, involving orthogonal polynomials, *Linear Algebra Appl.*, **52 / 52** : 293 – 300 (1983).

[GI] W. Gautschi and G.Inglese, Lower bounds for the condition number of Vandermonde matrix, *Numerische Mathematik*, **52** : 241 – 250 (1988).

[GF] I.Gohberg and I.Feldman, *Convolution equations and projection methods for their solutions*, Translations of Mathematical Monographs, **41**, Amer. Math. Soc., 1974.

[GK1] I.Gohberg and I.Koltracht, Efficient algorithm for Toeplitz plus Hankel matrices, *Integral Equations and Operator Theory*, **12** : 136 – 142 (1989).

[GK2] I.Gohberg and I.Koltracht, Mixed, componentwise and structured condition numbers, *SIAM J. on Matrix Analysis*, **14**, No. 3 : 688 – 704 (1993).

[GKK] I.Gohberg, T.Kailath and I.Koltracht, Efficient solution of linear systems of equations with recursive structure, *Linear Algebra Appl.*, **80** : 81 - 113 (1986).

[GKO] I.Gohberg, T.Kailath and V.Olshevsky, *Fast Gaussian elimination with partial pivoting for matrices with displacement structure*, Math. of Computation **64** No. 212 (1995), 1557-1576.

[GL] G.Golub and C.Van Loan, *Matrix Computations*, second edition, John Hopkins U. P., Baltimore, 1989.

[GO1] I.Gohberg and V.Olshevsky, Fast algorithms with preprocessing for matrix times vector multiplication, *Journal of Complexity*, **10** (1994), 411-427.

[GO2] I.Gohberg, and V.Olshevsky, Fast inversion of Chebyshev–Vandermonde matrices, *Numerische Mathematik*, **67**, No. 1 : 71 – 92 (1994).

[GO3] I.Gohberg and V.Olshevsky, Complexity of multiplication with vectors for structured matrices, *Linear Algebra Appl.*, **202** : 163 – 192 (1994).

[GO4] I.Gohberg and V.Olshevsky, Fast state space algorithms for matrix Nehari and Nehari-Takagi interpolation problems, *Integral Equations and Operator Theory*, **20** No. 1 : 44 – 83 (1994).

[GO5] I.Gohberg and V.Olshevsky, *The fast generalized Parker-Traub algorithm for inversion of Vandermonde and related matrices*, submitted, 1995.

[Hig1] N.Higham, Fast solution of Vandermonde–like systems involving orthogonal polynomials, *IMA J. Numerical Analysis*, **8** : 473 – 486 (1988).

[Hig2] N.Higham, Stability analysis of algorithms for solving confluent Vandermonde-like systems, *SIAM J. Matrix Analysis and Appl.*, **11**, No. 1 : 23 – 41 (1990).

[He] G.Heinig, *Inversion of generalized Cauchy matrices and other classes of structured matrices*, in : Linear Algebra in Signal Processing (Proc. of IMA-92 Workshop on linear algebra in signal processing), IMA volumes in Mathematics and its Applications, vol. **69** (1995), 95 – 114.

[HJR] G.Heinig, P.Jankowski and K.Rost, Fast inversion of Toeplitz-plus-Hankel matrices, *Numerische Mathematik* **52** : 665 – 682 (1988).

[HR] G.Heinig and K.Rost, *Algebraic methods for Toeplitz-like matrices and operators*, Operator Theory, vol. 13, Birkauser Verlag, Basel, 1984.

[KarSz] S.Karlin and G.Szegö, On certain determinants whose elements are orthogonal polynomials. In *Gabor Szegö Collected papers* – volume 3 (1982), Birkhauser, Boston, 603 – 762.

[K] T.Kailath, Signal processing applications of some moment problems, in *Moments in Mathematics* (H.Landau, ed.), AMS, vol. **37**, Providence, 1987, 71 – 109.

[K1] G. Kowalewski, *Interpolation und genäherte Quadratur*, Teubner, Berlin, 1932.

[KKM] T.Kailath, S.Kung and M.Morf, Displacement ranks of matrices and linear equations, *J. Math. Anal. and Appl.*, **68** : 395-407 (1979).

[KO1] T.Kailath and V.Olshevsky, *Displacement structure approach to Chebyshev–Vandermonde and related matrices*, Integral Equations and Operator Theory, **22**, 1995, 65 – 92.

[KO2] T.Kailath and V.Olshevsky, *Associated polynomials, confederate matrices and fast inversion of Szego-Vandermonde matrices*, Preprint.

[KS1] T. Kailath and A.H.Sayed, Fast algorithms for generalized displacement structures, in *Recent advances in mathematical theory of systems, control, networks and signal processing II*, Proc. of the MTNS-91 (H.Kimura, S.Kodama, Eds.), Mita Press, Japan, 1992, 27 – 32.

[KS2] T. Kailath and A.H.Sayed, Displacement structure : theory and applications, SIAM Review, **37** No.3 (1995), 297-386.

[L] F.Lander, The Bezoutian and the inversion of Hankel and Toeplitz matrices (in Russian), *Matem. Issled.*, Kishinev, **9** (No. 2) : 69 – 87 (1974).

[LT] P.Lancaster and M.Tismenetzky, *Theory of matrices with applications*, 2nd ed., Academic press, Orlando, 1985.

[M] M.Morf, Doubling Algorithms for Toeplitz and Related Equations, in *ICASSP 80 ; Proc. of the Int. Conf. on Acoustics, Speech, and Signal Processing*, IEEE, Denver, 954–959 (1980).

[MB] J.Maroulas and S.Barnett, Polynomials with respect to a general basis. I. Theory, *J. of Math. Analysis and Appl.*, **72** : 177 -194 (1979).

[P] F.Parker, *Inverses of Vandermonde matrices*, Amer. Math. Monthly, 71 (1964), 410 - 411.

[RO] L.Reichel and G.Opfer, Chebyshev–Vandermonde systems, *Math. of Comp.*, **57** No. 196 : 703 – 721 (1991).

[S] I.Schur, Über potenzreihen die im Inneren des Einheitskreises beschränkt sind, *Journal für die Reine und Angewandte Mathematik*, **147** : 205 – 232 (1917). English translation : in *Operator Theory : Advances and Applications* (I.Gohberg. ed.), vol. **18**, 31 – 88, Birkhäuser, Boston, 1986.

[SLAK] A.H.Sayed, H.Lev-Ari and T.Kailath, Fast triangular factorization of the sum of quasi-Toeplitz and quasi-Hankel matrices, *Linear Algebra and Appl.*, **191** : 77 – 106 (1993).

[Tr] J.Traub, Associated polynomials and uniform methods for the solution of linear problems, *SIAM Review*, **8** : 277 – 301 (1966).

[Ty] E.Tyrtyshnikov, How bad are Hankel matrices, *Numerische Mathematik*, **67**, No. 2 : 261 – 269 (1994).

[VS] L.Verde-Star, Inverses of generalized Vandermonde matrices, *J. Math. Anal. Appl.*, **131** : 341 – 353 (1988).

Submitted October 1994
Revised May 1996