



Displacement structure approach to Cauchy and Cauchy–Vandermonde matrices: inversion formulas and fast algorithms [☆]

Zheng-Hong Yang^{a,b,*}, Yong-Jian Hu^b

^aDepartment of Mathematics, P.O. Box 71, East Campus, China Agricultural University, 100083 Beijing, China

^bDepartment of Mathematics, Beijing Normal University, 100875 Beijing, China

Received 2 September 2000; received in revised form 22 February 2001

Abstract

The confluent Cauchy and Cauchy–Vandermonde matrices are considered, which were studied earlier by various authors in different ways. In this paper, we use another way called displacement structure approach to deal with matrices of this kind. We show that the Cauchy and Cauchy–Vandermonde matrices satisfy some special type of matrix equations. This leads quite naturally to the inversion formulas and fast algorithms for matrices of this kind. © 2002 Elsevier Science B.V. All rights reserved.

MSC: 15A57; 65D05

Keywords: Cauchy–Vandermonde matrix; Hermite rational interpolation; Displacement structure; Inversion formula; Fast algorithm

1. Introduction and preliminaries

Let

$$c = (c_1, \dots, c_n) \quad \text{and} \quad d = (d_1, \dots, d_k) \tag{1.1}$$

be two arrays of nodes with all c_i and d_j distinct pairwise. Associated with c and d the (simple) Cauchy matrix is defined as

$$C(c, d) = \left[\frac{1}{c_i - d_j} \right]_{i,j=1}^{n,k}, \tag{1.2}$$

[☆] Partially Supported by National Science Foundation of China.

* Correspondence address: Department of Mathematics, P.O. Box 71, East Campus, China Agricultural University, 100083 Beijing, China.

and the Cauchy–Vandermonde (CV) matrix is defined as

$$C_m(c, d) = [C(c, d) \quad V_m(c)], \tag{1.3}$$

where

$$V_m(c) = \begin{bmatrix} 1 & c_1 & \cdots & c_1^{m-1} \\ 1 & c_2 & \cdots & c_2^{m-1} \\ \vdots & \vdots & \cdots & \vdots \\ 1 & c_n & \cdots & c_n^{m-1} \end{bmatrix} \in C^{n \times m} \tag{1.4}$$

is the (simple) Vandermonde matrix corresponding to c .

One of the most important properties of Cauchy–Vandermonde matrices is their application to rational interpolation with fixed poles. For a given array of interpolation data (c_i, η_i) , we want to find a proper rational function $f(x) = q(x)/p(x)$ ($\deg q(x) < \deg p(x)$) with $p(x) = \prod_{j=1}^k (x - d_j)$ such that

$$f(c_i) = \eta_i, \quad (i = 1, \dots, n). \tag{1.5}$$

Let $f(x)$ have a partial fraction decomposition of the form $f(x) = \sum_{j=1}^k \xi_j / (x - d_j)$; then, the above problem is equivalent to solving the following system of equations:

$$C(c, d)\xi = \eta, \tag{1.6}$$

with $\xi = (\xi_1, \dots, \xi_k)^T$ and $\eta = (\eta_1, \dots, \eta_n)^T$.

If $f(x)$ is not proper, then the Euclidean algorithm yields

$$f(x) = \sum_{j=1}^k \frac{\xi_j}{x - d_j} + \xi_{k+1} + \xi_{k+2}x + \cdots + \xi_{k+m}x^{m-1}$$

and the above interpolation problem is equivalent to solving the linear system of equations

$$C_m(c, d)\xi = \eta \tag{1.7}$$

with $\xi = (\xi_1, \dots, \xi_{k+m})^T$ and η as above.

Cauchy and Cauchy–Vandermonde matrices appeared in a sequence of recent papers [9–11,1,3]. In [10,9] Mühlbach derived the determinant and inverse representations and the Lagrange–Hermite interpolation formula for Cauchy–Vandermonde matrix. In [11] Vavřín gave the factorization and inversion formulas for Cauchy and CV matrices in another way. The method in [10] is direct computation and seems to be rather complicated. Although in [11] the relations between Cauchy–Vandermonde matrices and rational interpolation problems were pointed out, the algorithm process for interpolants was not given.

In the present paper we make use of another method called displacement structure approach to deal with Cauchy and CV matrices. We point out that confluent Cauchy and CV matrices satisfy some of the special matrix equations and deriving their inverses is equivalent to solving only two linear systems of equations with Cauchy and CV matrices as coefficient matrices, and we also give the fast algorithms for solving such kinds of linear systems.

The concept of displacement structure was first introduced by Kailath et al. [5,6] for Toeplitz and Hankel matrices using the Stein-type operator $\nabla(\cdot)$ given by

$$\nabla(R) = R - FRA. \tag{1.8}$$

For details of displacement structure theory we refer the reader to the survey article of Kailath and Sayed [7]. In general, the generalized displacement operator is defined by

$$\nabla_{\{\Omega, \Delta, F, A\}}(R) = \Omega R \Delta - FRA, \tag{1.9}$$

where Ω, Δ, F and A are specified matrices such that $\nabla_{\{\Omega, \Delta, F, A\}}(R)$ has low rank, say r , independent of n . Then R is said to be a *structured* or a *displacement structure* matrix with respect to the displacement operator defined by (1.9), and r is referred to as the *displacement rank* of R . A special case of (1.9) will have a more simple form called an equation of Sylvester-type

$$\nabla_{\{\Omega, A\}}(R) = \Omega R - RA \tag{1.10}$$

which was first studied by Heinig [4] for Cauchy-like matrices.

Indeed, the Cauchy matrix $C(c, d)$ is the unique solution of the following Sylvester-type matrix equation:

$$D(c)C(c, d) - C(c, d)D(d) = \begin{bmatrix} 1 \\ 1 \\ \vdots \\ 1 \end{bmatrix} [1 \ 1 \ \dots \ 1], \tag{1.11}$$

where $D(c) = \text{diag}(c_i)_{i=1}^n$ and $D(d) = \text{diag}(d_j)_{j=1}^k$ are the diagonal matrices corresponding to c and d , respectively.

Eq. (1.11) implies that the Cauchy matrix $C(c, d)$ has displacement rank 1 with respect to the operator $\nabla_{\{D(c), D(d)\}}(\cdot)$. When $n=k$, multiplying Eq. (1.11) by $C(c, d)^{-1}$ from both left- and right-hand sides we get

$$D(d)C(c, d)^{-1} - C(c, d)^{-1}D(c) = -C(c, d)^{-1} \begin{bmatrix} 1 \\ 1 \\ \vdots \\ 1 \end{bmatrix} [1 \ 1 \ \dots \ 1]C(c, d)^{-1}. \tag{1.12}$$

If we denote by $u = (u_1, \dots, u_n)^T$ and $v = (v_1, \dots, v_n)^T$ the solutions of the following two equations:

$$C(c, d)u = [1 \ \dots \ 1]^T \quad \text{and} \quad v^T C(c, d) = [1 \ \dots \ 1], \tag{1.13}$$

then Eq. (1.12) by elementwise comparison shows that

$$C(c, d)^{-1} = - \left[\frac{u_i v_j}{d_i - c_j} \right]_{i,j=1}^n = -D(u)C(d, c)D(v), \tag{1.14}$$

where $D(u) = \text{diag}(u_i)_{i=1}^n$ and $D(v) = \text{diag}(v_j)_{j=1}^n$, which implies that computation of the inverse of the Cauchy matrix $C(c, d)$ is equivalent to solving only two systems of equations and one system in the symmetric case.

2. Confluent Cauchy and Cauchy–Vandermonde matrices: displacement structures and inversion formulas

In this section, we shall derive the displacement structures and fast inversion formulas for confluent Cauchy and CV matrices. Let us begin by recalling the definitions of confluent Cauchy and Cauchy–Vandermonde matrices. Our definitions are in accordance with those in [11] and differ slightly from those in [10]. Let

$$c = (\underbrace{c_1, \dots, c_1}_{n_1}, \underbrace{c_2, \dots, c_2}_{n_2}, \dots, \underbrace{c_p, \dots, c_p}_{n_p}) \tag{2.1}$$

and

$$d = (\underbrace{d_1, \dots, d_1}_{k_1}, \underbrace{d_2, \dots, d_2}_{k_2}, \dots, \underbrace{d_q, \dots, d_q}_{k_q}) \tag{2.2}$$

be two sequences of interpolation nodes with all c_i, d_j distinct pairwise and $\sum_{i=1}^p n_i = n, \sum_{j=1}^q k_j = k$. With c and d we associate two classes of matrices.

Definition 1. We call the $n \times k$ block matrix

$$C(c, d) = (C_{ij})_{i,j=1}^{p,q} \tag{2.3}$$

confluent Cauchy matrix corresponding to c and d , where

$$C_{ij} = (C_{ij}^{st})_{s=0,t=0}^{n_i-1,k_j-1} \in C^{n_i \times k_j}$$

and

$$C_{ij}^{st} = \frac{1}{s!t!} \frac{\partial^{s+t}}{\partial x^s \partial y^t} \left[\frac{1}{x-y} \right]_{y=d_j}^{x=c_i} = \binom{s+t}{s} \frac{(-1)^s}{(c_i - d_j)^{s+t+1}}$$

Definition 2. We call the $n \times (k + m)$ block matrix

$$C_m(c, d) = [C(c, d), V_m(c)] \quad (m \geq 0) \tag{2.4}$$

confluent Cauchy–Vandermonde matrix corresponding to c and d , where

$$V_m(c) = \begin{bmatrix} V_m(c_1) \\ \vdots \\ V_m(c_p) \end{bmatrix}, \quad \text{with } V_m(c_i) = \left[\binom{t}{s} c_i^{t-s} \right]_{s=0,t=0}^{n_i-1,m-1}, \tag{2.5}$$

is the confluent Vandermonde matrix of size $n \times m$ corresponding to c .

When $m = 0$ we assume that $C_0(c, d) = C(c, d)$.

As in the simple case, confluent Cauchy and CV matrices correspond to rational interpolation problems with prescribed possibly repeated poles. Indeed, let $p(x) = \prod_{j=1}^q (x - d_j)^{k_j}$ and

$$f(x) = \frac{q(x)}{p(x)} = \sum_{j=1}^q \sum_{t=0}^{k_j-1} \frac{\xi_{jt}}{(x - d_j)^{t+1}} + \sum_{j=1}^m \xi_j x^{j-1}.$$

Then the interpolation problem

$$\left. \frac{1}{s!} \frac{d^s f(x)}{dx^s} \right|_{x=c_i} = \eta_{is}$$

($i = 1, \dots, p, s = 0, \dots, n_i - 1$) is reduced to solve the following system:

$$C_m(c, d)\xi = \eta \tag{2.6}$$

where

$$\xi = \begin{bmatrix} \text{col}(\text{col}(\xi_{jt})_{t=0}^{k_j-1})_{j=1}^q \\ \text{col}(\xi_j)_{j=1}^m \end{bmatrix} \quad \text{and} \quad \eta = \text{col}[\text{col}(\eta_{is})_{s=0}^{n_i-1}]_{i=1}^p.$$

Hereafter, $\text{col}(a_i)$ denotes the column vector with a_i as components.

2.1. Displacement structures

In this subsection we derive the displacement structures for confluent Cauchy and CV matrices. Our starting point is the following Lemma.

Lemma 3. *Suppose that $J(\lambda)$ and $J(\mu)$ are the $m \times m$ and $n \times n$ lower triangular Jordan blocks with $\lambda \neq \mu$, respectively, and*

$$\Gamma = (\Gamma_{st})_{s=0, t=0}^{m-1, n-1}$$

is any $m \times n$ matrix. Then the Sylvester matrix equation

$$J(\lambda)X - XJ(\mu)^T = \Gamma \tag{2.7}$$

has the unique solution $X = (X_{st})_{s=0, t=0}^{m-1, n-1}$ given by

$$X_{st} = \sum_{\alpha=0}^s \sum_{\beta=0}^t \Gamma_{s-\alpha, t-\beta} \binom{\alpha + \beta}{\alpha} \frac{(-1)^\alpha}{(\lambda - \mu)^{\alpha+\beta+1}}. \tag{2.8}$$

Note that when $\lambda \neq \mu$, the uniqueness of the solution of Eq. (2.7) is a well-known fact since $J(\lambda)$ and $J(\mu)$ have disjoint spectra. If λ and μ have negative and positive real parts, respectively, then the solution of Eq. (2.7) is explicitly given by

$$X = - \int_0^{+\infty} e^{J(\lambda)t} \Gamma e^{-J(\mu)^T t} dt \tag{2.9}$$

(see, e.g., [8, Theorem 12.3.3]). In this case, straightforward calculation shows that the X in (2.9) is the same as in (2.8). But here we point out that the assertion is also true for any pair of $\lambda \neq \mu$.

Proof of Lemma 3. Denoting by S_k the $k \times k$ lower shift matrix

$$S_k = \begin{bmatrix} 0 & 0 & \cdots & \cdots & 0 \\ 1 & 0 & \cdots & \cdots & 0 \\ \vdots & \ddots & \ddots & & \vdots \\ 0 & 0 & \cdots & 1 & 0 \end{bmatrix}, \tag{2.10}$$

we then have $J(\lambda) = \lambda I + S_m$, $J(\mu) = \mu I + S_n$, and Eq. (2.7) is equivalent to $(\lambda - \mu)X_{st} + X_{s-1,t} - X_{s,t-1} = \Gamma_{st}(s = 0, \dots, m - 1, t = 0, \dots, n - 1)$. Therefore, we have recursive relations

$$X_{st} = (\lambda - \mu)^{-1}(\Gamma_{st} + X_{s,t-1} - X_{s-1,t})$$

with initial conditions $X_{-1,t} = X_{s,-1} = 0$. On the other hand, X_{st} in (2.8) can be rewritten as

$$\begin{aligned} X_{st} &= (\lambda - \mu)^{-1}\Gamma_{st} + \sum_{\alpha=0}^s \sum_{\beta=1}^t \Gamma_{s-\alpha,t-\beta}(-1)^\alpha \binom{\alpha + \beta}{\alpha} (\lambda - \mu)^{-\alpha-\beta-1} \\ &\quad + \sum_{\alpha=1}^k \Gamma_{s-\alpha,t}(-1)^\alpha (\lambda - \mu)^{-\alpha-1}. \end{aligned}$$

Therefore, it is sufficient to prove that X_{st} in (2.8) satisfies the following equality:

$$\begin{aligned} &\sum_{\alpha=0}^s \sum_{\beta=1}^t \Gamma_{s-\alpha,t-\beta}(-1)^\alpha \binom{\alpha + \beta}{\alpha} (\lambda - \mu)^{-\alpha-\beta} \\ &\quad + \sum_{\alpha=1}^s \Gamma_{s-\alpha,t}(-1)^\alpha (\lambda - \mu)^{-\alpha} + X_{s-1,t} - X_{s,t-1} = 0. \end{aligned} \tag{2.11}$$

If s resp. t in (2.8) be replaced by $s - 1$ resp. $t - 1$, we obtain

$$X_{s-1,t} = \sum_{\alpha=1}^s \sum_{\beta=0}^s \Gamma_{s-\alpha,t-\beta}(-1)^{\alpha-1} \binom{\alpha + \beta - 1}{\alpha - 1} (\lambda - \mu)^{-\alpha-\beta},$$

$$X_{s,t-1} = \sum_{\alpha=0}^s \sum_{\beta=0}^s \Gamma_{s-\alpha,t-\beta}(-1)^\alpha \binom{\alpha + \beta - 1}{\alpha} (\lambda - \mu)^{-\alpha-\beta}.$$

Substituting this into (2.11) shows that it is sufficient to prove the equality

$$\begin{aligned} & \sum_{\alpha=1}^s \sum_{\beta=1}^t \Gamma_{s-\alpha, t-\beta} \left\{ (-1)^\alpha \binom{\alpha + \beta}{\alpha} + (-1)^{\alpha-1} \binom{\alpha + \beta - 1}{\alpha - 1} - (-1)^\alpha \binom{\alpha + \beta - 1}{\alpha} \right\} \\ & \times (\lambda - \mu)^{-\alpha-\beta} + \sum_{\alpha=1}^s \Gamma_{s-\alpha, t} \{ (-1)^\alpha + (-1)^{\alpha-1} \} (\lambda - \mu)^{-\alpha} \\ & + \sum_{\beta=1}^t (\Gamma_{s, t-\beta} - \Gamma_{s, t-\beta}) (\lambda - \mu)^{-\beta} = 0. \end{aligned} \tag{2.12}$$

The last two terms in (2.12) are obviously equal to zero, and the basic combinatorial equality

$$\binom{\alpha + \beta}{\alpha} = \binom{\alpha + \beta - 1}{\alpha - 1} + \binom{\alpha + \beta - 1}{\alpha}$$

implies that the first term in (2.12) is also zero. This completes the proof. \square

Let Z be a column vector of length n and let Z be partitioned in accordance with the sequence of nodes c in (2.1) as $Z = \text{col}(Z_i)_{i=1}^p$ with $Z_i = \text{col}(Z_{is})_{s=0}^{n_i-1}$; Similarly, let $Y \in C^k$ be partitioned in accordance with d in (2.2) as $Y = \text{col}(Y_j)_{j=1}^q$ with $Y_j = \text{col}(Y_{jt})_{t=0}^{k_j-1}$. Also, let $z = \text{col}(z_i)_{i=1}^p$ with $z_i = [1 \ 0 \ \dots \ 0]^T \in C^{n_i}$ and $y = \text{col}(y_j)_{j=1}^q$ with $y_j = [1 \ 0 \ \dots \ 0]^T \in C^{k_j}$ be two fixed vectors. With Z and Y we associate block diagonal matrices

$$L(Z) = \text{diag}[L(Z_i)]_{i=1}^p, \tag{2.13}$$

where

$$L(Z_i) = \begin{bmatrix} Z_{i0} & 0 & \dots & 0 \\ Z_{i1} & Z_{i0} & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ Z_{i, n_i-1} & \dots & Z_{i1} & Z_{i0} \end{bmatrix} \in C^{n_i \times n_i}$$

is the lower triangular Toeplitz matrix with Z_i as its first column and $L(Y) = \text{diag}[L(Y_j)]_{j=1}^q$ is defined similarly. Following [2], we define the generalized Vandermonde matrix by

$$V_m(c, Z) = [Z, J(c)Z, \dots, J(c)^{m-1}Z], \tag{2.14}$$

where $J(c) = \text{diag}[J(c_i)]_{i=1}^p$ and $J(c_i)$ is the lower Jordan block of size $n_i \times n_i$ corresponding to c_i ; $J(d)$ is defined similarly. It can be easily shown that

$$V_m(c) = V_m(c, z) = [z, J(c)z, \dots, J(c)^{m-1}z]. \tag{2.15}$$

As an application of Lemma 3 we obtain the following theorem, which is very useful in the analysis below.

Theorem 4. Let Z and Y be given as above. Then the matrix equation

$$J(c)X - XJ(d)^T = ZY^T \tag{2.16}$$

has the unique solution

$$X = L(Z)C(c, d)L(Y)^T. \tag{2.17}$$

Proof. Indeed, if X is partitioned as $X = (X_{ij})_{i,j=1}^{p,q}$ with $X_{ij} = (X_{ij}^{st})_{s=0,t=0}^{n_i-1,k_j-1}$, then Eq. (2.16) is equivalent to the following series of equations:

$$J(c_i)X_{ij} - X_{ij}J(d_j)^T = Z_i Y_j^T, \tag{2.18}$$

($i = 1, \dots, p, j = 1, \dots, q$). Applying Lemma 3 to Eq. (2.18) we obtain

$$X_{ij}^{st} = \sum_{\alpha=0}^s \sum_{\beta=0}^t Z_{i,s-\alpha} \binom{\alpha + \beta}{\alpha} \frac{(-1)^\alpha}{(c_i - d_j)^{\alpha+\beta+1}} Y_{j,t-\beta}.$$

Rewritten in matrix form, this leads to

$$X_{ij} = L(Z_i)C_{ij}L(Y_j)^T,$$

which is equivalent to formula (2.17). This completes the proof. \square

Corollary 5. The confluent Cauchy matrix $C(c, d)$ defined as in Definition 1 satisfies the Sylvester-type matrix equation

$$J(c)C(c, d) - C(c, d)J(d)^T = zy^T. \tag{2.19}$$

In other words, the confluent Cauchy matrix $C(c, d)$ has displacement rank 1 with respect to $\nabla_{\{J(c), J(d)^T\}}(\cdot)$.

Proof. The proof is trivial since $L(z) = \text{diag}(L(z_i))_{i=1}^p = \text{diag}(I_{n_i})_{i=1}^p = I_n$, and $L(y) = \text{diag}(L(y_j))_{j=1}^q = \text{diag}(I_{k_j})_{j=1}^q = I_k$. \square

By introducing the matrix of order $k + m$

$$J_m(d, y) = \begin{bmatrix} J(d) & ye_-^T \\ O & S_m^T \end{bmatrix}, \tag{2.20}$$

where $e_- = [1 \ 0 \ \dots \ 0]^T \in C^m$ and y, S_m are defined as above, we are led to the following.

Theorem 6. The confluent CV matrix $C_m(c, d)$ defined as in Definition 2 satisfies the Sylvester-type matrix equation

$$J(c)C_m(c, d) - C_m(c, d)J_m(d, y)^T = J(c)^m ze_+^T, \tag{2.21}$$

where $e_+ = [0 \ \dots \ 0 \ 1]^T \in C^{k+m}$.

In other words, $C_m(c, d)$ had displacement rank 1 with respect to $\nabla_{\{J(c), J_m(d, y)^T\}}(\cdot)$.

Proof. Indeed,

$$\begin{aligned} & J(c)C_m(c, d) - C_m(c, d)J_m(d, y)^T \\ &= [J(c)C(c, d) - C(c, d)J(d)^T - V_m(c)e_-y^T, J(c)V_m(c) - V_m(c)S_m]. \end{aligned} \tag{2.22}$$

Since $V_m(c)e_-y^T = zy^T$, then by Theorem 5, the first term in (2.22) is equal to zero, and also according to (2.15) the second term

$$J(c)V_m(c) - V_m(c)S_m = [0 \ \cdots \ 0 \ J(c)^m z].$$

Summing up, the assertion follows immediately.

2.2. Inversion formulas

Let $J_1 = \text{diag}(J_{n_i})_{i=1}^p$, where

$$J_{n_i} = \begin{pmatrix} 0 & \cdots & \cdots & 1 \\ 0 & \cdots & 1 & 0 \\ \vdots & & & \vdots \\ 1 & 0 & \cdots & 0 \end{pmatrix}$$

is the $n_i \times n_i$ antiidentity matrix. In the same way, $J_2 = \text{diag}(J_{k_j})_{j=1}^q$ is defined. Note that

$$J_1 J(c) J_1 = J(c)^T, \quad J_1^T = J_1^{-1} = J_1,$$

and similarly, $J_2 J(d) J_2 = J(d)^T, \quad J_2^T = J_2^{-1} = J_2.$

Theorem 7. Let $C(c, d)$ be the confluent Cauchy matrix in Definition 1 and $n = k$, and let x, ω be the solutions of equations

$$[C(c, d)J_2]x = z \quad \text{and} \quad \omega^T [J_1 C(c, d)] = y^T, \tag{2.23}$$

respectively. Then the inverse of $C(c, d)$ is given by

$$C(c, d)^{-1} = -J_2 L(x) C(d, c) L(\omega)^T J_1. \tag{2.24}$$

where $L(x)$ and $L(\omega)$ are defined as in (2.13).

Proof. Multiplying Eq. (2.19) by J_1 from the left and by J_2 from the right, we get

$$J(c)^T [J_1 C(c, d) J_2] - [J_1 C(c, d) J_2] J(d) = (J_1 z)(J_2 y)^T.$$

Hence,

$$\begin{aligned} & J(d) [J_1 C(c, d) J_2]^{-1} - [J_1 C(c, d) J_2]^{-1} J(c)^T \\ &= - [J_2 C(c, d)^{-1} z] [y^T C(c, d)^{-1} J_1] = -x\omega^T. \end{aligned}$$

Then by Theorem 4 we have

$$[J_1 C(c, d) J_2]^{-1} = -L(x) C(d, c) L(\omega)^T$$

or equivalently,

$$C(c, d)^{-1} = -J_2 L(x) C(d, c) L(\omega)^T J_1.$$

This completes the proof. \square

Remark 1. Theorem 7 shows that the inverse of a confluent Cauchy matrix is almost of the same type of matrix (up to two block diagonal factors), and that it can be reduced to solve only two fundamental systems of linear equations with $C(c, d)$ as coefficient matrix. Therefore, inversion formula (2.24) yields a fast inverse.

Following [3], for convenience by $C(d, c, x, \omega)$ we denote the matrix as

$$C(d, c, x, \omega) = L(x) C(d, c) L(\omega)^T. \tag{2.25}$$

Theorem 8. Let $C_m(c, d)$ be defined as in Definition 2 and $n = k + m$, and let

$$x = \begin{bmatrix} x' \\ x'' \end{bmatrix} \quad \text{and} \quad \omega \in C^n$$

($x' \in C^k, x'' = \text{col}(X_j)_{j=1}^m \in C^m$) be the solutions of the following two equations:

$$C_m(c, d)x = J(c)^m z \quad \text{and} \quad \omega^T C_m(c, d) = e_+^T, \tag{2.26}$$

respectively. Then the inverse of $C_m(c, d)$ is given by

$$C_m(c, d)^{-1} = \begin{bmatrix} -J_2 C(d, c, J_2 x', J_1 \omega) J_1 \\ HV_m(c, J_1 \omega)^T J_1 \end{bmatrix} \tag{2.27}$$

where $V_m(c, J_1 \omega)$ and $C(d, c, J_2 x', J_1 \omega)$ are defined as in (2.14) and (2.25), and

$$H = \begin{bmatrix} -X_2 & \cdots & -X_m & 1 \\ \vdots & & & \vdots \\ -X_m & & & \vdots \\ 1 & \cdots & \cdots & 0 \end{bmatrix}.$$

Proof. Multiplying Eq. (2.21) by $C_m(c, d)^{-1}$ from the left and right, we obtain

$$\begin{aligned} & J_m(d, y)^T C_m(c, d)^{-1} - C_m(c, d)^{-1} J(c) \\ &= - [C_m(c, d)^{-1} J(c)^m z] [e_+^T C_m(c, d)^{-1}] = -x \omega^T. \end{aligned} \tag{2.28}$$

Let $C_m(c, d)^{-1}$ be partitioned into the form

$$C_m(c, d)^{-1} = \begin{bmatrix} B' \\ B'' \end{bmatrix}, \tag{2.29}$$

where $B' \in C^{k \times n}$ and $B'' \in C^{m \times n}$; then

$$J(d)^T B' - B' J(c) = -x' \omega^T$$

and by the same way as in Theorem 7 we get

$$B' = -J_2 C(d, c, J_2 x', J_1 \omega) J_1. \tag{2.30}$$

Moreover, according to Eq. (2.28) we have

$$e_{-y}^T B' + S_m B'' - B'' J(c) = -x'' \omega^T$$

and hence

$$e_{-y}^T (B' J_1) + S_m (B'' J_1) - (B'' J_1) J(c)^T = -x'' (J_1 \omega)^T. \tag{2.31}$$

Let B_j denote the j th row of $B'' J_1$, then Eqs. (2.31) gives us the recurrence relations

$$B_{j-1} = B_j J(c)^T - X_j (J_1 \omega)^T, \quad j = 2, \dots, m. \tag{2.32}$$

From (2.29) and the second of Eqs. (2.26) it follows that $B_m = \omega^T J_1$. Taking this into account, Eq. (2.32) leads to

$$B'' J_1 = \text{col}(B_j)_{j=1}^m = H V_m(c, J_1 \omega)^T$$

or equivalently,

$$B'' = H V_m(c, J_1 \omega)^T J_1. \tag{2.33}$$

Substituting B' in (2.30) and B'' in (2.33) into (2.29) completes the proof. \square

3. Fast algorithms for solving Cauchy–Vandermonde systems

As we have seen in Sections 2 and 3, we have to solve two systems of equations with CV coefficient matrices [see, Eqs. (1.6), (1.7), (2.6), (2.23) and (2.26)] when we want to solve rational interpolation problems, thus deriving their inversion formulas. Since Cauchy and CV matrices satisfy a class of displacement structure equations [see, Eqs. (1.11), (2.19) and (2.21)], these equations are special cases of the following general Sylvester displacement equation:

$$\nabla_{\{\Omega_1, A_1\}}(R_1) = \Omega_1 R_1 - R_1 A_1 = G_1 B_1, \tag{3.1}$$

where Ω_1 and A_1 are lower and upper triangular matrices, $G_1 \in C^{n \times 1}$ and $B_1 \in C^{1 \times n}$. We may use the fast algorithm given in [2] to solve these linear systems. The idea of the algorithm is to derive the LU factorization quickly by using the displacement equations, which, therefore, is a generalized Gaussian elimination.

Lemma 9. [2] *Let the matrix*

$$R_1 = \begin{bmatrix} r_1 & u_1 \\ l_1 & R_{22}^{(1)} \end{bmatrix}$$

satisfy Eq. (3.1). If $r_1 \neq 0$, then the Schur complement $R_2 = R_{22}^{(1)} - (1/r_1)l_1u_1$ satisfies the following equation:

$$\Omega_2 R_2 - R_2 A_2 = G_2 B_2, \tag{3.2}$$

where Ω_2 and A_2 are obtained from Ω_1 and A_1 by deleting the first row and column, respectively, and

$$\begin{bmatrix} 0 \\ G_2 \end{bmatrix} = G_1 - \begin{bmatrix} 1 \\ \frac{1}{r_1} l_1 \end{bmatrix} g_1, \quad [0 \ B_2] = B_1 - b_1 \begin{bmatrix} 1 & \frac{1}{r_1} u_1 \end{bmatrix}, \tag{3.3}$$

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

Since R_1 has the triangular factorization of the form

$$R_1 = \begin{bmatrix} r_1 & u_1 \\ l_1 & R_{22}^{(1)} \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ \frac{1}{r_1} l_1 & I \end{bmatrix} \begin{bmatrix} r_1 & u_1 \\ 0 & R_2 \end{bmatrix} = LU,$$

applying Lemma 9 we may write down an implementation of LU factorization for R_1 as below:

1. *Compute the first row $[r_1 \ u_1]$ and the first column $\begin{bmatrix} r_1 \\ l_1 \end{bmatrix}$ of R_1 , respectively, which is equivalent to solving two triangular linear systems*

$$[r_1 \ u_1](\omega_{11}I - A_1) = g_1 B_1, \quad (\Omega_1 - a_{11}I) \begin{bmatrix} r_1 \\ l_1 \end{bmatrix} = G_1 b_1;$$

(ω_{11} and a_{11} are the (1,1) entry of Ω_1 and A_1 , respectively).

2. *Write down the first column $\begin{bmatrix} 1 \\ \frac{1}{r_1} l_1 \end{bmatrix}$ of L and first row $[r_1 \ u_1]$ of U .*
3. *Compute the generator $\{G_2, B_2\}$ for the Schur complement R_2 by using (3.3).*
4. *Repeat processes 1–3 for the displacement equation (3.2) R_2 satisfying.*

The overall complexity of the above algorithm is $O(n^2)$ arithmetic operations. Finally, we point out that we may consider block LU factorization when c or d has multiple nodes. In this case, the first step of the above algorithm is modified to solve two block triangular linear systems; accordingly, the second step is modified to write down the first n_1 columns of L and first n_1 rows of U .

4. Example

In the simple nodes case: $n_i = 1$, $c = (c_1, \dots, c_n)$ and $k_j = 1$, $d = (d_1, \dots, d_n)$

$$C(c, d) = \begin{bmatrix} (c_1 - d_1)^{-1} & (c_1 - d_2)^{-1} & \dots & (c_1 - d_n)^{-1} \\ (c_2 - d_1)^{-1} & (c_2 - d_2)^{-1} & \dots & (c_2 - d_n)^{-1} \\ \vdots & \vdots & \dots & \vdots \\ (c_n - d_1)^{-1} & (c_n - d_2)^{-1} & \dots & (c_n - d_n)^{-1} \end{bmatrix}.$$

According to the algorithm given in Section 3, we may write down the k th column L_k of L and the k th row U_k of U in the LU decomposition of $C(c, d)$:

$$L_k = \begin{bmatrix} 0 \\ \vdots \\ 0 \\ 1 \\ \prod_{j=1}^k \frac{c_k - d_j}{c_{k+1} - d_j} \prod_{i < k} \frac{c_{k+1} - c_i}{c_k - c_i} \\ \vdots \\ \prod_{j=1}^k \frac{c_k - d_j}{c_n - d_j} \prod_{i < k} \frac{c_n - c_i}{c_k - c_i} \end{bmatrix},$$

$$U_k = \left[0 \ \dots \ 0 \ \frac{1}{c_k - d_k} \prod_{i < k} \frac{(c_k - c_i)(d_k - d_i)}{(c_k - d_i)(d_k - c_i)} \ \dots \ \frac{1}{c_k - d_n} \prod_{i < k} \frac{(c_k - c_i)(d_n - d_i)}{(c_k - d_i)(d_n - c_i)} \right]$$

and the generators of Schur complement R_k :

$$G_k = \begin{bmatrix} \prod_{i < k} \frac{c_k - c_i}{c_k - d_i} \\ \vdots \\ \prod_{i < k} \frac{c_n - c_i}{c_n - d_i} \end{bmatrix} \in C^{n+1-k},$$

$$B_k = \left[\prod_{i < k} \frac{d_k - d_i}{d_k - c_i} \ \dots \ \prod_{i < k} \frac{d_n - d_i}{d_n - c_i} \right].$$

Here we assume that $\prod_{i < k} = 1$ when $k = 1$.

References

[1] M. Gasca, J.J. Martinez, G. Mühlbach, Computation of rational interpolants with prescribed poles, J. Comput. Appl. Math. 26 (1989) 297–309.

- [2] I. Gohberg, T. Kailath, V. Olshevsky, Fast Gaussian elimination with partial pivoting for matrices with displacement structure, *Math. Comput.* 64 (1995) 1557–1576.
- [3] G. Heinig, Generalized Cauchy–Vandermonde matrices, *Linear Algebra Appl.* 270 (1998) 45–77.
- [4] G. Heinig, K. Rost, *Algebraic Methods for Toeplitz-like Matrices and Operators*, Operator Theory, Vol. 13, Birkhauser, Basel, 1984.
- [5] T. Kailath, Remarks on the origin of the displacement rank concept, *Appl. Math. Comput.* 45 (1991) 193–206.
- [6] T. Kailath, S. Kung, M. Morf, Displacement ranks of matrices and linear equations, *J. Math. Anal. Appl.* 68 (1979) 395–407.
- [7] T. Kailath, A.H. Sayed, Displacement structure: theory and applications, *SIAM Rev.* 37 (3) (1995) 297–386.
- [8] P. Lancaster, M. Tismenetsky, *The Theory of Matrices with Applications*, 2nd Edition, Academic Press, New York, 1985.
- [9] G. Mühlbach, Computation of Cauchy–Vandermonde determinants, *J. Number Theory* 43 (1993) 74–81.
- [10] G. Mühlbach, On Hermite interpolation by Cauchy–Vandermonde systems: the Lagrange formula, the adjoint and the inverse of Cauchy–Vandermonde matrix, *J. Comput. Appl. Math.* 67 (1996) 147–159.
- [11] Z. Vavřín, Confluent Cauchy and Cauchy–Vandermonde matrices, *Linear Algebra Appl.* 258 (1997) 271–293.