J. Korean Math. Soc. **42** (2005), No. 1, pp. 1–16

A FAST FACTORIZATION ALGORITHM FOR A CONFLUENT CAUCHY MATRIX

Kyungsup Kim

Abstract. This paper presents a fast factorization algorithm for confluent Cauchy-like matrices. The algorithm consists of two parts. First, a confluent Cauchy-like matrix is transformed into a Cauchy-like matrix available to pivot without changing its structure. Second, a fast partial pivoting factorization algorithm for the Cauchy-like matrix is presented. A new displacement structure cannot possibly generate all entries of a transformed matrix, which is called by "partially reconstructible". This paper also discusses how the proposed factorization algorithm can be generally applied to partially reconstructive matrices.

1. Introduction

The generalized Nevanlinna-Pick interpolation problem has been investigated, which can be encountered in several applications including model reduction, sensitivity minimization and robust stabilization [2, 4]. Specially, we note factorization algorithms for the corresponding Pick matrix. Several fast algorithms for the special structure of the Pick matrix like a Toeplitz-like and Cauchy-like matrix with only $O(n^2)$ arithmetic operation have been introduced [6, 8, 4, 7, 11]. However, $O(n^3)$ arithmetic operations are needed to compute the triangular LU factorization for a general dense matrix [6].

For an $n \times n$ Toeplitz matrix R, the number of operations of the triangular factorization $R = LDL^*$ can be reduced to $O(n^2)$ from $O(n^3)$ if the structural property is used, i.e., Schur algorithm[6, 8, 4, 7, 11]. Here, L is a lower triangular matrix, D is a diagonal matrix, and A^* is the complex conjugate transposition of A. A matrix is said to have the

Received March 12, 2003.
2000 Mathematics Subject Classification: Primary 15A23, 65T50.
Key words and phrases: matrix factorization, fast algorithm, FFT.
This work was supported by the Post-doctoral Fellowship Program of Korea Science & Engineering Foundation (KOSEF).

displacement structure introduced in [9] if it obeys the general displacement equation of the form

$$R - FRF^* = GJG^*, \quad F \in \mathbb{C}^{n\times n}, \quad G \in \mathbb{C}^{n\times\alpha}, \tag{1}$$

where F is a lower triangular matrix, and a signature matrix J is defined such that the diagonal entries of J are 1 or -1, otherwise zero. Here, G is called a generator. A triple pair, (F, G, J), is called a generator of R. The rank of G is called the displacement rank of R. Toeplitz, Cauchy and Confluent Cauchy matrices have displacement structures (1) with special generators (F, G, J) [7, 9, 10]. The computational burden can be reduced to $O(n^2)$, if R has the lower displacement rank than n as the Toeplitz or Cauchy matrix [7, 9], while the triangular factorization for a dense matrix can be solved by the Gauss elimination method [6], which requires $O(n^3)$ multiplication operations.

It is known that Schur algorithm often produces very inaccurate results for indefinite matrices because of the recursion nature of the algorithm, which sequentially processes all leading submatrices. It can break down if one of leading submatrices is singular or nearly singular. In addition, even if one encounters a nonsingular submatrix, near-break-downs occur for ill-conditioned submatrices. In order to avoid the singular problem, pivoting techniques are used [6]. But, pivoting matrix with a special displacement structure, the desired displacement structure to implement a fast algorithm can destroyed. Therefore, the merit of fast algorithm is reduced.

From this motivation, this paper presents a fast pivoting factorization algorithm for a confluent Cauchy-like matrix. A confluent Cauchy-like matrix is transformed into a Cauchy-like matrix via a matrix with a circulant displacement structure. We consider several problems that appear when Schur algorithms are applied to the derived Cauchy-like matrix.

The paper is structured as follows. In Section 2, confluent Cauchy-like matrices are introduced. Section 3 shows how a confluent Cauchy-like matrix can be transformed into a Cauchy-like matrix. Section 4 discusses how a pivoting generalized Schur algorithm is applied to Cauchy-like matrices. Numerical experiment results are presented in section 5, and a conclusion follows.

2. Generalized Nevanlinna-pick interpolation problems

Let us review the Nevanlinna-pick interpolation problems. It is assumed that distinct points $\{\zeta_i\}$ in open unit disk $\mathbb{D}$ for $1 \le i \le p$ exist. A generalized Nevanlinna-pick interpolation determines a rational function $K(z)$ satisfying the following conditions

A. $K(z)$ is analytic inside open unit disk,
B. $K(z)$ is passive on the boundary $\partial\mathbb{D}$ of unit disk, i.e., $\sup_{z\in\partial\mathbb{D}} |K(z)| < 1$.
C. $K(z)$ satisfies tangential confluent interpolation conditions such that for $k = 1, 2, \ldots, p$ and $j = 1, 2, \ldots, m_k$, a pair $(x_{k,j}, y_{k,j})$ satisfies the following relation:

$$x_{k,j}\frac{K^{j-1}(\zeta_k)}{(j-1)!} = y_{k,j}.$$

These interpolation problems can be solved via a 2×2 rational matrix function, $W(z)$, with a state-space representation [2] as follows:

$$W(z) = \begin{bmatrix} W_{11}(z) & W_{12}(z) \\ W_{21}(z) & W_{22}(z) \end{bmatrix} = I + G(zI - F)^{-1}R^{-1}G^*. \tag{2}$$

We note that R is a certain structured matrix. Set $\mathrm{diag}(\mathrm{A},\mathrm{B}) = \begin{bmatrix} A & O \\ O & B \end{bmatrix}$ and $J = \mathrm{diag}(1, -1)$. A Pick matrix R in (2) satisfies a displacement structure (1) (see [10]) where F, and G are defined by

$$F = \mathrm{diag}(\mathcal{J}_{\zeta_1}, \mathcal{J}_{\zeta_2}\ldots, \mathcal{J}_{\zeta_p}), \quad G = \begin{bmatrix} x_{1,1} & y_{1,1} \\ \vdots & \vdots \\ x_{1,m_1} & y_{1,m_1} \\ x_{2,1} & y_{2,1} \\ \vdots & \vdots \\ x_{p,m_p} & y_{p,m_p} \end{bmatrix}, \tag{3}$$

where $\sum_{k=1}^{p} m_k = n$ and an $m_k \times m_k$ lower triangular Jordan block $\mathcal{J}_{\zeta_k}$ is defined such that the diagonal entries of $\mathcal{J}_{\zeta_k}$ are ζ_k, $(i, i-1)$ entries for $2 \le i \le m_k$ are 1, and otherwise zero. A Pick matrix R satisfying (1) with (3) is called a *confluent Cauchy-like matrix.* All solutions $K(z)$ for the interpolation problem are parameterized as follows

$$K(z) = [W_{11}(z)U(z) + W_{12}(z)][W_{21}(z)U(z) + W_{22}]^{-1}$$

where $U(z)$ is an arbitrary rational function satisfying the conditions A and B. However, in order to solve (2), one needs to compute $R^{-1}G$. The arithmetic complexity of the matrix multiplication $R^{-1}G$ is $O(n^3)$. A Pick matrix R is a Toeplitz-like matrix if F is a single Jordan block with

$\zeta_1 = 0$, and a Pick matrix R is a Cauchy-like matrix if F is a diagonal matrix with $m_k = 1$ for $1 \leq k \leq p$. However, we are more interested in the case where $m_k >> 1$. Assume that a Pick matrix R not is positive definite. If the condition A is relaxed, then R is not positive definite. Then due to the roundoff error, the Schur algorithm can give inaccurate results [8].

The equation in (1) is called a discrete time Lyapunov equation or a Stein equation [2]. If F is a lower triangular matrix with a lower bandwidth 2, then all entries of R can be computed within the order of $O(n^2)$. The pseudocode solving the stein equation is given as follows:

ALGORITHM 2.1 (Solving Stein equation). Assume that an $n \times n$ lower triangular matrix F satisfies a lower bandwidth of 2 and $1 - f_{ii} f_{jj}^* \neq 0$ for $1 \leq i, j \leq n$ where f_{ij} is an (i, j) entry of F. Compute R with (1) as follows:

Input: A generator $(G,\ F,\ J)$ of R.
Output: R
Procedure:

$C = GJG^*$
$R(:,1) = (I - F \cdot f_{11}^*)^{-1} \cdot C(:,1)$
for $k = 2 : n$
  $R(:,k) = (I - F \cdot f_{kk}^*)^{-1} \{C(:,k) + F \cdot R(:,k-1)\}$
end

Let us count the arithmetic operation order of Algorithm 2.1. Since $I - F \cdot f_{kk}$ for each $k = 1, \ldots, n$ and F are nonsingular and lower triangular matrices having a lower bandwidth 2, for a given x, the operation orders of $(I - F \cdot f_{kk})^{-1} \cdot x$ and $F \cdot x$ are $O(2n)$ (see, e.g. [6]). The total arithmetic operation order of Algorithm 1 is approximately $2n^2$.

3. A transformation into Cauchy-like matrices

Let us consider some properties of a circulant matrix to give a motivation. A circulant matrix is defined by $\mathrm{Circ}(r)$ for any

$$r = \begin{bmatrix} r_0^* & r_1^* & \cdots & r_{n-1}^* \end{bmatrix}^*$$

such that

$$\mathrm{Circ}(r) = \begin{bmatrix} r_0 & r_{n-1} & \cdots & r_1 \\ r_1 & r_0 & \cdots & r_2 \\ \vdots & & & \\ r_{n-1} & r_{n-2} & \cdots & r_0 \end{bmatrix}. \tag{4}$$

For any circulant matrix $Circ(r)$, a diagonal matrix $\mathcal{D}$ is derived by

$$\mathcal{D} = \mathcal{F}\mathrm{Circ}(r)\mathcal{F}^* \quad \text{with} \quad \mathcal{F} = \frac{1}{\sqrt{n}}(w^{kl})_{k,l=0}^{n-1}, \tag{5}$$

for a given $\omega = e^{\frac{-2\pi i}{n}}$ with $i = \sqrt{-1}$, and a circulant matrix $\mathrm{Circ}(r)$ (see [8] [5] and references therein). $\mathcal{F}$ is called a Fast Fourier Transform (FFT) matrix, which means including a fast algorithm and a discrete Fourier transform. A solution R of (1) is called as a *circulant displacement structure* matrix if F is a circulant matrix $F = \mathrm{Circ}(r)$.

Now, consider a generalized Cauchy matrix. A generalized Cauchy matrix R is of the form

$$R = \left[\frac{c_i J c_j^*}{1-d_i d_j^*}\right]_{1\le i,j\le n} \quad (c_i \in \mathbb{C}^{1\times\alpha}) \tag{6}$$

with an $\alpha \times \alpha$ signature matrix J. A matrix R is denoted by a *Cauchy-like matrix* if R satisfies a displacement structure such that

$$R - \mathcal{D}R\mathcal{D}^* = GJG^* \tag{7}$$

with $\mathcal{D} = \mathrm{diag}(d_1, d_2, \ldots, d_n)$, and $G = \begin{bmatrix} c_1^* & \cdots & c_n^* \end{bmatrix}^*$. Cauchy-like matrices have some good properties that can derive a fast algorithm with pivoting. The key motivation point is that a Cauchy-like matrix retains the same displacement structure as (7) after symmetry permutations for the columns and rows are performed, i.e., it allows pivoting technique to be incorporated into a Schur algorithm for the factorization of a generalized Cauchy-like matrices [8].

Initially, a single Jordan block is considered as a simple case. Suppose that F is a single $n \times n$ Jordan block $F = \mathcal{J}_\zeta$ with $|\zeta| \neq 1$. For the case, it does not permit a pivoting technique since the necessary assumption that F is a lower triangular matrix is destroyed after pivoting. The next theorem shows how a displacement structure of a confluent Cauchy-like matrix with a Jordan block F can be transformed into a new circulant displacement structure.

THEOREM 3.1. *Assume that a Hermitian matrix $R \in \mathbb{C}^{n\times n}$ is a solution of the displacement equation*

$$R - \mathcal{J}_\zeta R \mathcal{J}_\zeta^* = GJG^* \tag{8}$$

where $\mathcal{J}_\zeta$ is an $n \times n$ Jordan block for a given ζ with $|\zeta| \neq 1$, a generator $G \in \mathbb{C}^{n\times\alpha}$ has a low rank, J is an $\alpha \times \alpha$ signature matrix and C_ζ is defined by

$$C_\zeta = \mathrm{Circ}(r) \tag{9}$$

with a given $r = \begin{bmatrix}1 & \zeta^* & 0 & \cdots & 0\end{bmatrix}^* \in \mathbb{C}^n$. *Then* R *satisfies a new displacement equation*

$$R - C_\zeta R C_\zeta^* = \bar{G}_1 J_1 \bar{G}_1^*, \quad J_1 = \begin{bmatrix} J & O \\ O & \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix} \end{bmatrix}, \tag{10}$$

where $\bar{G}_1 = \begin{bmatrix} G & y_1 & y_2 \end{bmatrix}$ *is a matrix with less rank than* $\alpha + 2$. *Here,* y_i *is a linear combination of* $\{E_1, y\}$ *where* $y = \mathcal{J}_\zeta R E_n$, $r_{i,j}$ *is an* (i,j) *entry of* R *and* E_i *is an* i *standard basis.*

Proof. The circulant matrix C_ζ is rewritten by $C_\zeta = J_\zeta + E_1 E_n^*$. This term is inserted into the left side of (10). From (8), we have

$$\begin{aligned} R - C_\zeta R C_\zeta^* &= R - \mathcal{J}_\zeta R \mathcal{J}_\zeta^* - (E_1 E_n^* R \mathcal{J}_\zeta^* + \mathcal{J}_\zeta R E_n E_1^* + r_{nn} E_1 E_1^*) \\ &= GJG^* - (E_1 y^* + y E_1^* + r_{nn} E_1 E_1^*). \end{aligned} \tag{11}$$

Since R is a Hermitian matrix, the diagonal entries of R are real numbers. Hence, a function $\operatorname{sign}(x)$ can be defined such that $\operatorname{sign}(x) = 1$ for $x \geq 0$ and $\operatorname{sign}(x) = -1$ for $x < 0$. The last three terms in (11) can be rewritten as

$$E_1 y^* + y E_1^* + r_{nn} E_1 E_1^* = \begin{cases} \operatorname{sign}(r_{nn})\{(\tau E_1 - \frac{y}{\tau})(\tau E_1 - \frac{y}{\tau})^* - \frac{y}{\tau}\frac{y}{\tau}^*\} & \text{if } r_{nn} \neq 0 \\ \frac{(E_1+y)}{\sqrt{2}}\frac{(E_1+y)}{\sqrt{2}}^* - \frac{(E_1-y)}{\sqrt{2}}\frac{(E_1-y)}{\sqrt{2}}^* & \text{otherwise.} \end{cases}$$

where $\tau = \sqrt{|r_{nn}|}$. Therefore, a new generator $\bar{G}_1$ can be written by

$$\bar{G}_1 = \begin{cases} \begin{bmatrix} G & (\tau E_1 - \frac{y}{\tau}) & \frac{y}{\tau} \end{bmatrix} & \text{if } r_{nn} \neq 0 \text{ and } \operatorname{sign}(r_{nn}) > 0 \\ \begin{bmatrix} G & \frac{y}{\tau} & (\tau E_1 - \frac{y}{\tau}) \end{bmatrix} & \text{if } r_{nn} \neq 0 \text{ and } \operatorname{sign}(r_{nn}) < 0 \\ \begin{bmatrix} G & \frac{(E_1-y)}{\sqrt{2}} & \frac{(E_1+y)}{\sqrt{2}}) \end{bmatrix} & \text{otherwise.} \end{cases} \tag{12}$$

Thus, the rank of a generator $\bar{G}_1$ derived from (12) is shown to be less than $\alpha + 2$. □

The above theorem shows how a displacement structure of a simple confluent Cauchy-like matrix with a single Jordan block, $F = \mathcal{J}_\zeta$, can be transformed into a matrix with a circulant displacement structure. A generalized version will be developed for the case where a Jordan blocks matrix F is defined by (3).

THEOREM 3.2. *Assume that a matrix* R *satisfies the displacement structure (1) where* $F = \operatorname{diag}(\mathcal{J}_{\zeta_1}, \ldots, \mathcal{J}_{\zeta_p})$, G *is an* $n \times \alpha$ *matrix, and* J *is an* $\alpha \times \alpha$ *signature matrix. The initial values are defined by* $F_0 = F$,

$G_0 = G$, $J_0 = J$ and $\alpha_0 = \alpha$. Assume that for some $k > 0$, a matrix R satisfies a displacement structure such that

$$R - F_k R F_k^* = \bar{G}_k J_k \bar{G}_k^* \tag{13}$$

where F_k is defined by

$$F_k = \operatorname{diag}(C_{\zeta_1}, \cdots, C_{\zeta_k}, \mathcal{J}_{\zeta_{k+1}}, \cdots, \mathcal{J}_{\zeta_p}) \tag{14}$$

for circulant matrices $\{C_{\zeta_i}\}$ and Jordan block matrices $\{\mathcal{J}_{\zeta_i}\}$, $\bar{G}_k \in \mathbb{C}^{n\times\alpha_k}$ is a generator, and J_k is a signature matrix. Define a submatrix dimension by $\mu_k = \sum_i^k m_i$. Then F_k, $\bar{G}_k$, J_k and α_k can satisfy the following recurrence relations:

$$F_{k+1} = F_k + E_{\mu_k+1} \cdot E^*_{\mu_{k+1}} \tag{15}$$
$$\bar{G}_{k+1} = \begin{bmatrix}\bar{G}_k & y_{k,1} & y_{k,2}\end{bmatrix} \tag{16}$$
$$J_{k+1} = \operatorname{diag}\{J_k, 1, -1\} \tag{17}$$
$$\alpha_{k+1} = \alpha_k + 2, \tag{18}$$

where $y_{k,1}$ and $y_{k,2}$ are proper linear combinations of $\{y, E_{\mu_k+1}\}$ when y is defined by $y = F_k \cdot R \cdot E_{\mu_{k+1}}$ and E_j is a j standard basis. Moreover, $\operatorname{rank}(\bar{G}_k) \le \operatorname{rank}(G) + 2k$.

Proof. The equation $F_{k+1} = F_k + E_{\mu_k+1} \cdot E^*_{\mu_{k+1}}$ in (15) is directly derived by the definition of F_k. Insert F_{k+1} (15) into $R - F_{k+1}RF^*_{k+1}$.

$$R - F_{k+1}RF^*_{k+1} = \bar{G}_k J \bar{G}_k - E_{\mu_k+1}y - yE^*_{\mu_k+1} - r_{\mu_{k+1},\mu_{k+1}} E_{\mu_k} E_{\mu_k}$$

can be derived. Since the derivation of $\bar{G}_{k+1}$ is similar to $\bar{G}_1$ (12) in Theorem 3.1, the relation of the generator (16) and the relation of the signature matrix (17) can be derived directly. Since $\operatorname{rank}(\bar{G}_{k+1}) \le \operatorname{rank}(\bar{G}_k) + 2$, the last statement is trivial by the induction. □

Using Theorem 3.2 recursively until $k = p$, R satisfies a displacement structure where the diagonal sub-blocks of a matrix F_p (i.e. for $k = p$ in (15)) are circulant matrices C_{ζ_k}. R has a *generalized circulant displacement structure*, if all the diagonal sub-blocks of F are circulant matrices and the others are zero, i.e., $F = F_p$ in (15).

Next, we show how to transform a generalized circulant displacement structured matrix into a Cauchy-like matrix. Note that $\mathcal{F}\mathrm{Circ}(r)\mathcal{F}^*$ is a diagonal matrix where $\mathcal{F}$ is an FFT matrix. The transformation from more general confluent Cauchy-like matrices into Cauchy-like matrices is summarized in the next theorem.

THEOREM 3.3. *Assume that a confluent Cauchy-like matrix R is a solution of the displacement equation where $F = \mathrm{diag}(\mathcal{J}_{\zeta_1}, \ldots, \mathcal{J}_{\zeta_p})$, G is an $n \times \alpha$ matrix, J is an $\alpha \times \alpha$ signature matrix and all ζ_k's are $|\zeta_k| \neq 1$. Then a confluent Cauchy-like matrix R can be transformed into a Cauchy-like matrix $\hat{R}$ such that $\hat{R} = \mathcal{F}R\mathcal{F}^*$ is a solution of the displacement equation*

$$\hat{R} - \mathcal{D}\hat{R}\mathcal{D}^* = \hat{G}J_p\hat{G}^*, \quad \hat{G} = \mathcal{F}\bar{G}_p, \tag{19}$$

where $\mathcal{F}$ is defined by $\mathcal{F} = \mathrm{diag}\{\mathcal{F}_1, \ldots, \mathcal{F}_p\}$, $\mathcal{F}_k$'s are $m_k \times m_k$ dimensional FFT matrices, a diagonal matrix $\mathcal{D}$ is defined by $\mathcal{D} = \mathcal{F}F_p\mathcal{F}^*$. $\bar{G}_p$, F_p, and J_p are defined in Theorem 3.2.

Proof. By using Theorem 3.2, a new displacement structure with a block circulant matrix F_p can be defined in (13) when $k = p$. Multiply by $\mathcal{F}^*$ and $\mathcal{F}$ on the left and right, respectively, of $R - F_pRF_p^* = \bar{G}_pJ\bar{G}_p^*$. It can be see that $\mathcal{F}R\mathcal{F}^* - \mathcal{F}F_p\mathcal{F}^*\mathcal{F}R\mathcal{F}^*\mathcal{F}F_p^*\mathcal{F}^* = \mathcal{F}\bar{G}_pJ\bar{G}_p^*\mathcal{F}^*$ since $\mathcal{F}^*\mathcal{F}$ is a identity matrix. Set $\hat{R} = \mathcal{F}R\mathcal{F}^*$, $\mathcal{D} = \mathcal{F}F_p\mathcal{F}^*$. From this, it is relatively straightforward to show that a solution $\hat{R}$ of the displacement equation $\hat{R} - \mathcal{D}\hat{R}\mathcal{D}^* = \hat{G}J_p\hat{G}^*$ (19) is a Cauchy-like matrix since $\mathcal{D} = \mathcal{F}F_p\mathcal{F}^*$ is a diagonal matrix. □

4. A factorization algorithm for a confluent Cauchy matrix

This section proposes a fast partial pivoting factorization for a Hermitian confluent Cauchy-like matrix R. It was shown how a confluent Cauchy-like matrix is transformed into a Cauchy-like matrix. A fast factorization algorithm for the Cauchy-like matrix using the generalized Schur algorithm is introduced in [8]. This section discusses how to apply the generalized Schur algorithm to a Cauchy-like matrix derived from a confluent Cauchy-like matrix.

If $\det(F - I) = 0$, then a classical Schur algorithm can break down [6, 11]. We consider a generalized Schur algorithm, which is different from a classical Schur algorithm of the simple form. The following theorem from [1, 8] is a basis of our new generalized Schur algorithm.

THEOREM 4.1 (Generalized Schur algorithm). *Let F be a block lower triangular matrix and R be a Hermitian solution of the equation* (1) *for an $n \times \alpha$ matrix G and a signature matrix J. Choose γ such that* $\det(\gamma \mathrm{I} - \mathrm{F}) \neq 0$ *and $|\gamma| = 1$. Let the matrices be partitioned by*

$$F = \begin{bmatrix} F_{11} & 0 \\ F_{21} & F_{22} \end{bmatrix}, \quad R = \begin{bmatrix} R_{11} & R_{12} \\ R_{21} & R_{22} \end{bmatrix}, \quad G = \begin{bmatrix} G_{11} \\ G_{21} \end{bmatrix}. \tag{20}$$

If the submatrix R_{11} *is invertible, then the Schur complement* R_1 *of* R

$$R_1 = R_{22} - R_{21} R_{11}^{-1} R_{12} \tag{21}$$

satisfies a displacement structure as follows:

$$R_1 - F_{22} R_1 F_{22}^* = G_1 J G_1^* \tag{22}$$

$$G_1 = G_{21} - ((\gamma I - F_{22}) R_{21} R_{11}^{-1} - F_{21})(\gamma I - F_{11})^{-1} G_{11}. \tag{23}$$

Proof. Refer to Theorem 4.3 in [1] or Lemma 4.1 in [8]. □

The assumption $\det(\gamma \mathrm{I} - \mathrm{F}) \neq 0$ is needed in order to prevent a breakdown of the algorithm. From Theorem 4.1, it can be seen that a new generator (F_1, G_1, J) of the Schur complement R_1 of R can be computed from the generator (F, G, J) of R, R_{11}, R_{21} and R_{21}.

We consider what happens when a confluent Cauchy-like matrix is transformed into a Cauchy-like matrix. For a given $F = \mathcal{J}_\zeta \in \mathbb{C}^{n \times n}$ and a given $n \times n$ FFT matrix $\mathcal{F}$, a k diagonal entry $\{d_k\}$ of $\mathcal{D} = \mathcal{F} C_\zeta \mathcal{F}^*$ is a root of the characteristic polynomial $P(z) = (z - \zeta)^n - 1$, which is expressed by $d_k = \zeta + w^k$ for a given $\omega = e^{\frac{-2\pi i}{n}}$. In the result, the diagonal matrix $\mathcal{D}$ defined in (19) can be computed without computational complexity.

In the result of Theorem 4.1, some block submatrixces such as R_{11} and R_{21} in (20) should be computed from G. Assume that an (i, j) entry $\hat{r}_{ij}$ of $\hat{R}$ is reconstructible, i.e., $1 - d_i d_j^* \neq 0$ and $[\eta_{ij}] = \hat{G} J_p \hat{G}^*$. Then $\hat{r}_{ij}$ can be computed by

$$\hat{r}_{ij} = \frac{\eta_{ij}}{1 - d_i d_j^*} \tag{24}$$

for a j-th diagonal entry d_j of $\mathcal{D}$ with $1 - d_i d_j^* \neq 0$. In the case where $d_i d_j^* - 1 = 0$ for some i, j, an (i, j) entry $\hat{r}_{ij}$ of $\hat{R}$ in (19) is not uniquely determined from the data of $\hat{G}$, J_p, and $\mathcal{D}$. It is referred to as the *partially reconstructible* matrix introduced in [8]. Therefore, a unique solution $\hat{R}$ cannot be determined by (24). The next stage is to determine how to construct an entry $\hat{r}_{ij}$ of $\hat{R}$ when $\hat{r}_{ij}$ is not reconstructible. An unreconstructible set $\mathcal{R}$ from $\mathcal{D}, \hat{G}$ and J_p can be defined by

$$\mathcal{R} = \{(\hat{r}_{i,j}, i, j) | d_i \cdot d_j^* - 1 = 0\}. \tag{25}$$

Because of the assumption $|\zeta_k| \neq 1$, a Pick matrix R that is a solution of (1) with the condition (3) is reconstructive. Therefore the element $(\hat{r}_{i,j}, i, j)) \in \mathcal{R}$ can be computed by $\hat{r}_{i,j} = (\mathcal{F}R\mathcal{F}^*)_{i,j}$. While $\hat{r}_{ij}$ cannot be obtained from $\hat{G}$, the entries of the Cauchy-like matrix derived from a confluent Cauchy-like matrix can be computed. For each recursion step, elements of $\mathcal{R}$ must be updated by explicitly computing the Schur complement as (21), since they cannot be obtained from the generator. However, the computation load increases, if the number of $\mathcal{R}$ increases.

The next stage considers a pivoting problem in the generalized Schur algorithm. A single step of the standard Cholesky factorization is valid if the upper left block $\hat{R}_{11}$ of $\hat{R}$ is invertible. Now, to enhance the accuracy of the computations, the Gaussian elimination is proceeded by symmetric row and column permutations such that

$$\hat{R} \leftarrow P\hat{R}P^* \tag{26}$$

where P is a permutation matrix. If $\hat{R}$ has a Cauchy-like matrix as in (21), then a new updated matrix $\hat{R}$ has a displacement structure matrix with a generator $(P\mathcal{D}P^*, P\hat{G}, J_p)$. A Cauchy-like scheme acts on $\hat{R}$ after permutations, while for general displacement structure matrices the result of the permutation destroys the structure (e.g., the assumption that F is a lower triangular matrix cannot be satisfied). The Bunch-Kaufman algorithm [3, 6] is often used for symmetry indefinite matrices. A Bunch-Kaufman algorithm can be modified for a Cauchy-like matrix.

ALGORITHM 4.2. Let $\hat{R}$ be a solution of the displacement structure equation in (19) and $\mathcal{R}$ be an unreconstructible set of $\hat{R}$ defined as (25).

Input: A generator $(\mathcal{D}, \hat{G}, J_p)$ and an unreconstructible set $\mathcal{R}$ of $\hat{R}$.
Output: $(\mathcal{D}, \hat{G}, J_p)$, a permutation matrix P, m, and $\mathcal{R}$.
Procedure:

· Compute the first column $\hat{R}(:,1)$ of $\hat{R}$ from $\hat{G}$ for reconstructible case or take them from $\mathcal{R}$
· $\nu = \frac{1+\sqrt{17}}{8}$
· $\epsilon = |\hat{r}_{t,1}| = \max\{|\hat{r}_{2,1}, \ldots, |\hat{r}_{n,1}|\}$
if $\epsilon \neq 0$
  if $|\hat{r}_{1,1}| \leq \nu\epsilon$
    · $m = 1$; $P = I$
  else
    · Compute the t-th column $\hat{R}(:,t)$ of $\hat{R}$ from $\hat{G}$ for reconstructible case or take them from $\mathcal{R}$
    · $\sigma = |\hat{r}_{p,t}| = \max\{|\hat{r}_{1,t}|, \ldots, |\hat{r}_{t-1,t}|, |\hat{r}_{t+1,t}|, \ldots, |\hat{r}_{n,t}|\}$

if $\sigma|\hat{r}_{1,1}| \le \nu\epsilon^2$
  · $m = 1$; $P = I$
else if $|\hat{r}_{t,t}| \le \nu\sigma$
  · $m = 1$ and choose P such that $(P \cdot \hat{R} \cdot P^*)_{1,1} = \hat{r}_{t,t}$
else
  · $m = 2$ and choose P such that $(P \cdot \hat{R} \cdot P^*)_{2,1} = \hat{r}_{t,p}$
end
end
end
return $\hat{G} = P\hat{G}$; $\mathcal{D} = P\mathcal{D}P^*$; Reordering $\mathcal{R}$

Since the Bunch-Kaufman algorithm requires knowledge of the entries of one more column of $\hat{R}$ for each step, this pivoting technique gives the Schur algorithm the additional cost of $O(2\alpha n)$. A pseudocode of a fast partial pivoting factorization algorithm for Confluent Cauchy matrices is presented in the next algorithm. The algorithm is factorized in the following form:

$$P\mathcal{F}R\mathcal{F}^*P^* = LDL^* \tag{27}$$

where $\mathcal{F}$ is defined as Theorem 3.3, P is a permutate matrix, L is a lower triangular matrix and D is at most a triple diagonal matrix.

ALGORITHM 4.3. A factorization algorithms for Confluent Cauchy:

- **Input:** A generator (G, F, J) defined as Theorem 3.3.
- **Output:** $P\mathcal{F}R\mathcal{F}^*P^* = LDL^*$.
- **Procedure:**
  1. Compute a generator $(F_p, \bar{G}_p, J_p)$.
     - Compute $R(:, \mu_k)$ using Algorithm 2.1.
     - Compute a new generator $(F_p, \bar{G}_p, J_p)$ using Theorem 3.2.
  2. Compute a new generator $(\mathcal{D}, \hat{G}, J_p)$ and an unreconstructible set $\mathcal{R}$.
     - Compute a new generator $(\mathcal{D}, \hat{G}, J_p)$ using Theorem 3.3.
     - Compute an unreconstructible set $\mathcal{R}$ such that $\hat{r}_{i,j} = (\mathcal{F}R\mathcal{F}^*)_{i,j}$ for $1 - d_i d_j^* = 0$.
  3. Perform Theorem 4.1.

     Sub-procedure:
     · $L(:,:) = O$; $D(:,:) = O$;
     · ct =1;
     while $(ct < n)$
       · Using Algorithm 4.2, compute $\mathcal{R}$, P, m and $(\mathcal{D}, \hat{G}) \leftarrow (P\mathcal{D}P^*, P\hat{G})$

· Compute $\hat{R}(:,1:m)$ from a generator $(\mathcal{D},\hat{G},J_p)$

$$\hat{r}_{i,j} = \begin{cases} \hat{r}_{i,j} = \frac{\eta_{ij}}{1-d_i d_j^*} & \text{if } \hat{r}_{i,j} \text{ is reconstructible} \\ \text{Take } \hat{r}_{r,j} \text{ from } \mathcal{R} & \text{otherwise.} \end{cases}$$

· $L(ct:n, ct:ct+m-1) = \hat{R}(:,1:m)\hat{R}(1:m,1:m)^{-1}$
· $D(ct:ct+m-1, ct:ct+m-1) = \hat{R}(1:m,1:m)$
· Compute a generator $(\mathcal{D},\hat{G})$ of Schur complement by Theorem 4.1
· Update the elements of $\mathcal{R}$ by computing directly Schur complement
· $ct = ct + m$;
end

Let us analyze the complexity of computing a new algorithm. At the step (1) in Algorithm 4.3, a Stein equation is solved using Algorithm 2.1. Its operation order is approximately $2n^2$. At step (2), the entries of $\mathcal{D}$ are directly computed with an n order and depending on FFT multiplication the computation order of $\hat{G}$ is $\alpha_p n\log(n)$. Step (3) is $O(\alpha_p n^2)$. It can be seen that the total multiplication order of Algorithm 4.3 is proportional to n^2 if $\mathcal{R} = \emptyset$. However, there are two facts that can result in an increase of the computation load. First, when p increase, the proposed algorithm is not efficient since $\alpha_p = 2p+\alpha$. Therefore, $p \ll n$ is claimed. Second, if the element number ξ of $\mathcal{R}$ increases, then the computational complexity increases proportionally with a rate ξn.

5. Numerical experiments

The proposed algorithm was applied to factorizing some examples to clarify its benefits. A large amount of computer experiments were performed with the algorithm designed in this paper to compare it to other available algorithms. All the algorithms were implemented in double precision, for which the unit round-off was $2^{-56} = 1.4 \times 10^{-17}$. MATLAB was used to implement the proposed algorithm and the other available algorithms. The following abbreviations used in these examples are described in Table 1. Table 1 provides an approximation for computational complexities. Computational errors were evaluated by the componentwise matrix residual error

$$RE = \max_{(i,j)} |(LDL^* - P\mathcal{F}T\mathcal{F}P^*)_{(i,j)}|$$

TABLE 1. The arithmetic complexity for algorithms

Abbreviation	Full Name	The Complexity
R	Solving $R - FRF^* = GJG^*$	$O(2n^2)$
$\hat{G} = \mathcal{F}\bar{G}_p$	Fast Fourier Transform of G_p	$O(n \log(n))$
GE	Gaussian Elimination [6]	$O(n^3)$
GSA	Generalized Schur Algorithm [12]	$O(n^2)$
PSA	Proposed Schur algorithm	$O(n^2)$

to conform the accuracy of a new factorization algorithm. Example 5.1 provides an example for an irregular matrix that needs pivoting. In Example 5.2, the numerical complexity and accuracy of the three algorithms (GE, PSA and GSA) are compared for a simple Confluent Cauchy-like matrix. Furthermore, in Example 5.3, the numerical complexity and the accuracy are compared for a general Confluent Cauchy-like matrix.

EXAMPLE 5.1 (The irregular case). A simple structured matrix satisfying (3) is given such that

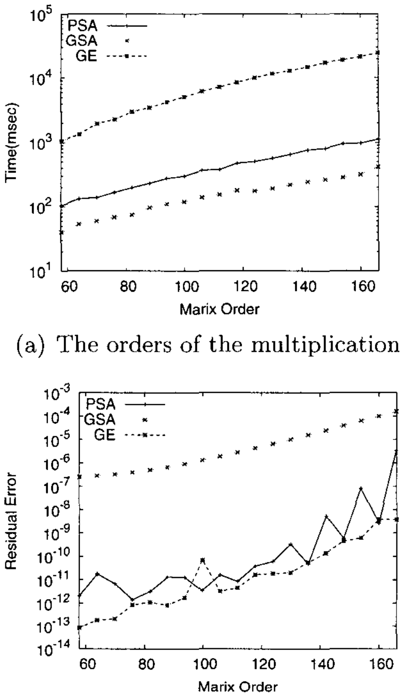
$$F = \begin{bmatrix} .1 & 0 & 0 \\ 1 & 0.1 & 0 \\ 0 & 1 & 0.1 \end{bmatrix}, \quad G = \begin{bmatrix} 1 & -1 \\ 1 & 0 \\ 1 & 0 \end{bmatrix}.$$

It can be seen that the $(1,1)$ entry of R is zero. The eigenvalues of R are $\{1.5932, 0.6096, -1.1737\}$. The condition number of R is 2.6134. Therefore, this case is irregular and indefinite. It can not be factorized without pivoting. GE and PSA can be applied but GSA cannot.

EXAMPLE 5.2 (The simple case $F = \mathcal{J}_\zeta$). A simple confluent Cauchy-like matrix satisfying the following condition such that R is a solution of a displacement equation (3) where F is a simple Jordan block defined by $F = \mathcal{J}_\zeta$ with $\zeta = 0.04$, and a generator matrix G that satisfies

$$G = \begin{bmatrix} 1 & 0.9999 \\ -1 & 0 \\ \vdots & \vdots \\ (-1)^{n-1} & 0 \end{bmatrix}.$$

After a transformation of a confluent Cauchy-like matrix into a Cauchy-like matrix, the rank of a new generator is 4. Let n be a dimension of R. PSA, GSA and GE were exercised from $n = 58$ to $n = 166$ with step



(a) The orders of the multiplication

(b) Residual errors

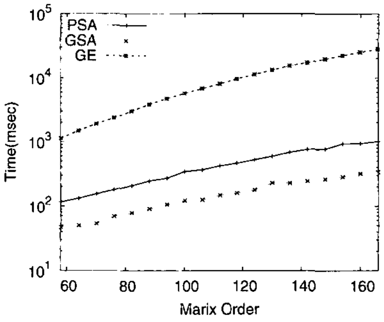
FIGURE 1. Comparison of PSA, GSA, and GE for simple confluent Cauchy-like matrices

size 6. Figure 1 shows that the PSA is faster than the GE and more accurate than the GSA.

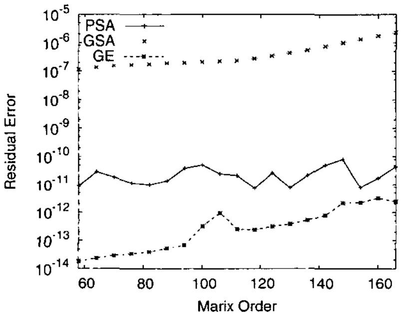
EXAMPLE 5.3 (The general case $F = \mathrm{diag}\{\mathcal{J}_{\zeta_1}, \mathcal{J}_{\zeta_2}\}$). A confluent Cauchy-like matrix R is a solution of a displacement equation (3) where F is $F = \mathrm{diag}(\mathcal{J}_{\zeta_1}, \mathcal{J}_{\zeta_2})$ with $\zeta_1 = 0.04$ and $\zeta_2 = 0.05$, and a generator matrix G that satisfies

$$G = \begin{bmatrix} 1 & 0.9999 \\ -1 & 0 \\ \vdots & \vdots \\ (-1)^{n-1} & 0 \end{bmatrix}.$$

After transforming a confluent Cauchy-like matrix into a Cauchy-like matrix, the dimension of a new generator is 6. Let n be a dimension of R. PSA, GSA and GE were exercised from $n = 58$ to $n = 166$ with a



(a) The orders of the multiplication



(b) Residual errors

FIGURE 2. Comparison of PSA, GSA, and GE for general confluent Cauchy-like matrices

step size of 6. Figure 2 shows that the PSA is faster than the GE and more accurate than the GSA.

In results, the proposed algorithm is faster than the GE. In accuracy, the proposed algorithm is better than the GSA without pivoting even though the proposed algorithm can be worse than the GE, which is not a fast algorithm.

6. Conclusion

This paper proposed a transformation of a Confluent Cauchy-like matrix into a Cauchy-like matrix via a matrix with a generalized circulant displacement structure. After transforming a confluent Cauchy-like matrix to a Cauchy-like matrix, we proposed a fast pivoting algorithm to factorize the derived Cauchy-like matrix. A method to avoid reconstructible property were discussed. Finally, Experiments were performed in order to clarify that this new algorithm is faster than Cholesky algorithm and more accurate than the generalized Schur algorithm.

ACKNOWLEDGEMENTS. Author thanks the anonymous reviewer for useful comments and suggestions.

References

[1] D. Alpay and H. Dym, *On a new class of reproducing kernel spaces and a new generalization of the Iohvidov laws*, Linear Algebra Appl. **178** (1993), 109–183.

[2] J. A. Ball, I. Gohberg, and L. Rodman, *Interpolation of Rational Matrix Functions*, Birkhauser Verlag, Basel, Switzerland, 1990.

[3] J. R. Bunch and L. Kaufman, *Some stable methods for calculating inertia and solving symmetric linear systems*, Math. Comp. **31** (1977), 162–179.

[4] Ph. Delsarte, Y. Genin and Y. Kamp, *On the role of the Nevanlinna-Pick Problem in circuit and system theory*, Int. J. Circuit Theory and Appl. **9** (1981), 177–187.

[5] P. Gader, *Displacement operator based decompositions of matrices using circulants or other group matrices*, Linear Algebra Appl. **139** (1990), 111–131.

[6] G. H. Golub and C. F. Van Loan, *Matrix Computations*, Johns Hopkins University Press, Baltimore, MD, 3rd edition, 1996.

[7] G. Heinig and K. Rost, *Algebraic Methods for Toepliz-like Matrices and Operators*, Birkhäuser-Verlag, Basel, 1984.

[8] T. Kailath and V. Olshevsky, *Bunch-Kaufman pivoting for partially reconstructable Cauchy-like matrices with application to Toeplitz-like matrices and to boundary rational matrix interpolation problems*, Linear Algebra Appl. **254** (1997), 251–302.

[9] S. Kung T. Kailath and M. Morf, *Displacement ranks of matrices and linear equations*, J. Math. Anal. Appl. **68** (1979), 395–407. also Bull. Amer. Math. Soc. **1** (1979), 799–773.

[10] V. Olshevsky and A. Shokrollahi, *A Superfast Algorithm for Confluent Rational Tangential Interpolation Problem via Matrix-vector Multiplication for Confluent Cauchy-like Matrices., in Structured Matrices in Mathematics, Computer Science and Engineering 1*, Contem. Math., **280** (2001), 32–46.

[11] I. Schur, *Über potenzreihen die in inneren des einheitskreises beschräkt sind*, Parts I and II, J. Reine Angew. Math. **147** (1917), 205–232.

[12] Z. H. Yang, and Y. J. Hub, *Displacement structure approach to Cauchy and Cauchy Vandermonde matrices: inversion formulas and fast algorithms*, J. Comput. Appl. Math. **138** (2002), no. 2, 259–272.

Department of Computer Science & Engineering
Chungnam National University
Taejon 305-764, Korea
E-mail: kskim@cs.cnu.ac.kr