Fast Recovery and Approximation of Hidden Cauchy Structure

Jörg Liesen* Robert Luce*

December 9, 2014

Abstract

We derive an algorithm of optimal complexity which determines whether a given matrix is a Cauchy matrix, and which exactly recovers the Cauchy points defining a Cauchy matrix from the matrix entries. Moreover, we study how to approximate a given matrix by a Cauchy matrix with a particular focus on the recovery of Cauchy points from noisy data. We derive an approximation algorithm of optimal complexity for this task, and prove approximation bounds. Numerical examples illustrate our theoretical results.

1 Introduction

A Cauchy matrix $C(s, t) \in \mathbb{C}^{m,n}$ is defined by $n+m$ data points $s \in \mathbb{C}^m$, $t \in \mathbb{C}^n$ satisfying $s_i \neq t_j$ for all $i, j$, the Cauchy points, via the relation

$$C(s, t) = [c_{ij}] := \begin{bmatrix} 1 \\ s_i - t_j \end{bmatrix}.$$ 

Cauchy matrices occur in numerous applications. To give just one example, let $(s_i, z_i) \in \mathbb{C} \times \mathbb{C}$ be given with pairwise distinct values $s_1, \ldots, s_n$ and let $t_1, \ldots, t_n \in \mathbb{C}$ be given with $s_i \neq t_j$ for all $i, j$. Then the coefficients $a = [a_1, \ldots, a_n]^T \in \mathbb{C}^n$ such that the rational function

$$r(s) = \sum_{j=1}^{n} \frac{a_j}{s - t_j}$$

satisfies $r(s_i) = z_i$, $i = 1, \ldots, n$, can be found by solving the linear system

$$C(s, t)a = z.$$
Note that the condition $s_i \neq t_j$ for the Cauchy points appears naturally in this application (as in many others) by the requirement that the poles of the rational function $r(s)$ must be distinct from the points where the (finite) values of $r(s)$ are prescribed.

A Cauchy matrix satisfies the Sylvester type displacement equation

$$SC(s, t) - C(s, t)T = 1_m1_n^T,$$

where $S := \text{diag}(s) \in \mathbb{C}^{m \times m}$, $T := \text{diag}(t) \in \mathbb{C}^{n \times n}$, and $1_m := [1, \ldots, 1]^T \in \mathbb{R}^m$. Hence the $(S,T)$-displacement rank of $C(s, t)$ is equal to 1. For more details on displacement structure and pointers to the relevant literature we refer to [3, Section 12.1]. Due to this special structure, several fast algorithms exist for performing matrix computations with $C(s, t)$. For example, an $LU$ decomposition of $C(s, t)$ with partial pivoting can be computed in $O(mn)$ [1] (the GKO algorithm), and matrix-vector products with $C(s, t)$ can be computed faster than $O(mn)$ [4] (the fast multipole method); see also [2] and [6, Sec. 3.6].

In this work we are, however, not concerned with performing computations with Cauchy matrices. Rather we study the problem of determining whether a given matrix $A \in \mathbb{C}^{m \times n}$ is equal or at least “near” to a Cauchy matrix. For such matrices we derive algorithms of optimal complexity that compute Cauchy points $s \in \mathbb{C}^m$, $t \in \mathbb{C}^n$ with $A = C(s, t)$ when $A$ is a Cauchy matrix, or with $A \approx C(s, t)$ when certain conditions are satisfied. We are not aware that a similar study has appeared in the literature before.

This cheap recognition (and approximation) could possibly be useful in black-box linear system solvers, where, instead of solving a given linear system using general purpose methods, the solver would first run the proposed algorithms to determine whether the given matrix is actually close to a Cauchy matrix, and to solve the system with a specialized algorithm in that case. Since this upfront test runs in time proportional to the size of the input, the implied overhead is negligible.

Let us briefly describe our general approach and the outline of this paper. When $A = [a_{ij}] = C(s, t)$ is a Cauchy matrix, but the corresponding Cauchy points $s, t$ are unknown, these can be computed by solving the $mn$ nonlinear equations (in $m + n$ variables)

$$\frac{1}{s_i - t_j} = a_{ij}, \quad 1 \leq i \leq m, \quad 1 \leq j \leq n. \quad (1)$$

For the Cauchy matrix $A$ we have $a_{ij} \neq 0$, and hence the equations (1) are equivalent to the $nm$ linear equations (in $m + n$ variables)

$$s_i - t_j = \frac{1}{a_{ij}}, \quad 1 \leq i \leq m, \quad 1 \leq j \leq n. \quad (2)$$

In Section 2 we discuss the linearization (2) of the equations (1) in more detail, study uniqueness properties of its solution and derive a $O(m + n)$ algorithm for solving (2).
If the given matrix $A = [a_{ij}]$ is not a Cauchy matrix, and the task is to approximate $A$ with a Cauchy matrix, one would ideally like to solve the nonlinear optimization problem

$$\min_{s,t} \sum_{i=1}^{m} \sum_{j=1}^{n} \left| \frac{1}{s_i - t_j} - a_{ij} \right|^2 = \min_{s,t} \|C(s,t) - A\|_F^2,$$  \hspace{1cm} (3)

where $\|\cdot\|_F$ denotes the Frobenius norm. Instead of solving (3) directly, we consider the ordinary (convex) linear least squares problem

$$\min_{s,t} \sum_{i=1}^{m} \sum_{j=1}^{n} \left| s_i - t_j - \frac{1}{a_{ij}} \right|^2 = \min_{s,t} \|D(s,t) - A^{-1}\|_F^2, \hspace{1cm} (4)$$

where

$$A^{-1} := [a_{ij}]^{-1}, \quad D(s,t) := [s_i - t_j] \in \mathbb{C}^{m,n}.$$  

The problem (4) can be considered a linearization of the nonlinear problem (3). We first show in Section 3.1 how to solve (4) to optimality in $O(mn)$. In Section 3.2 we relate the solutions obtained from (4) to solutions of the original problem (3). In particular, we analyze when optimal solutions of (4) deliver good approximations to the Cauchy points of a noisy Cauchy matrix $A = C(s,t) + N$, where the matrix $N$ represents some data error. We illustrate our results by numerical experiments.

We give concluding remarks and a few open questions in Section 4.

### 2 Exact recovery of Cauchy points

Let $A = [a_{ij}] \in \mathbb{C}^{m \times n}$ with $a_{ij} \neq 0$ for all $i, j$ be given. There exist Cauchy points $s \in \mathbb{C}^m$, $t \in \mathbb{C}^n$ with $A = C(s,t)$, i.e., $A$ is a Cauchy matrix, if and only if the equations (1) hold. Since $a_{ij} \neq 0$ for all $i, j$, the equations (1) are equivalent with the equations (2), and these can be written in matrix form as

$$U \begin{bmatrix} s \\ t \end{bmatrix} = b, \hspace{1cm} (5)$$

where

$$U := [I_m \otimes 1_n - 1_m \otimes I_n] \in \mathbb{C}^{mn \times (m+n)}, \quad b := \text{vec}(A^{-T}) \in \mathbb{C}^{mn}. \hspace{1cm} (6)$$

Here $A^{-T} := (A^{-1})^T = (A^T)^{-1}$, and vec denotes the standard vectorization operator of matrices.

Using the (overdetermined) linear system (5)–(6) we can test whether a given matrix $A = [a_{ij}]$ with $a_{ij} \neq 0$ for all $i, j$ is a Cauchy matrix or not. At first sight there are two possible cases for the latter: Either the system has a solution but it does not define Cauchy points, or the system has no solution. The equations (2) show that the first case cannot occur under our assumption $a_{ij} \neq 0$ for all
i, j, since for any solution \([s^*]\) of (5)–(6) the differences \(s_i - t_j\) must be nonzero for all \(i, j\). In other words, if \([s^*]\) solves (5)–(6) for a componentwise nonzero right hand side \(b\), then \(s, t\) are Cauchy points. There are, of course, matrices \(A\) with all entries nonzero for which no solution exists.

**Example 2.1.** For \(A = \begin{bmatrix} 1 & -1 \\ -1 & 1 \end{bmatrix}\) we have

\[
U = \begin{bmatrix} 1 & 0 & -1 & 0 \\ 1 & 0 & 0 & -1 \\ 0 & 1 & -1 & 0 \\ 0 & 1 & 0 & -1 \end{bmatrix}, \quad \text{vec}(A^{[-T]}) = \begin{bmatrix} 1 \\ -1 \\ -1 \\ 1 \end{bmatrix},
\]

and a simple computation shows that there exists no solution of the corresponding system (5)–(6). Hence \(A\) is not a Cauchy matrix.

If \(s \in \mathbb{C}^m, t \in \mathbb{C}^n\) are Cauchy points, then

\[
C(s, t) = C(s + \alpha 1_m, t + \alpha 1_n)
\]

for all \(\alpha \in \mathbb{C}\). Consequently, the Cauchy points \(s, t\) of a Cauchy matrix \(A = [a_{ij}]\) are not uniquely determined by the values \(a_{ij}\). We will show next that this global translation of the Cauchy points is the only source of ambiguity.

**Theorem 2.2.** The matrix \(U\) in (6) satisfies \(\ker(U) = \text{span}\{1_{m+n}\}\). Thus, if \([\tilde{s}]\) is a solution of (5)–(6), then the set of all solutions is given by

\[
\{[\tilde{s}] + \alpha 1_{m+n} \mid \alpha \in \mathbb{C}\}.
\]

**Proof.** Since \(U1_{m+n} = 0\) we have \(\text{span}\{1_{m+n}\} \subseteq \ker(U)\). If \(z = [\tilde{y}] \in \ker(U)\) with \(x \in \mathbb{C}^m\) and \(y \in \mathbb{C}^n\), then

\[
x_j 1_n = y, \quad j = 1, \ldots, m.
\]

In particular, \(y = x_1 1_n\), which implies \(x_j = x_1\) for \(j = 2, \ldots, m\), so that \(z = x_1 1_{m+n} \in \text{span}\{1_{m+n}\}\).

In order to remove the ambiguity about the possible Cauchy points that define a given Cauchy matrix we introduce the following definition.

**Definition 2.3.** Let \(A \in \mathbb{C}^{m,n}\) be a Cauchy matrix. We say that \(\tilde{s} \in \mathbb{C}^m, \tilde{t} \in \mathbb{C}^n\) are normalized Cauchy points for \(A\), if \(A = C(\tilde{s}, \tilde{t})\) and \(\| [\tilde{s}^T, \tilde{t}^T]^T \|_2\) is minimal among all possible Cauchy points \(s \in \mathbb{C}^m, t \in \mathbb{C}^n\) with \(A = C(s, t)\).

If \(A = C(s, t)\), then normalized Cauchy points for \(A\) can be found by solving the minimization problem

\[
\min_{\alpha \in \mathbb{C}} f(\alpha) := \| z - \alpha 1_{m+n} \|_2^2, \quad z := \begin{bmatrix} \tilde{s} \\ \tilde{t} \end{bmatrix}.
\]

(7)
Algorithm 1 Optimal recovery of normalized Cauchy points

Input: Cauchy matrix $A \in \mathbb{C}^{m,n}$, $a_{ij} \neq 0$ for all $i,j$.
Output: Normalized Cauchy points $\tilde{s}, \tilde{t}$ such that $A = C(\tilde{s}, \tilde{t})$.

1: $s(1) \leftarrow 0$ \{Choice arbitrary\}
2: $t(1:n) \leftarrow s(1) - A(1,1:n)^{-1}$
3: $s(2:m) \leftarrow t(1) + A(2:m,1)^{-1}$
4: $\alpha \leftarrow \frac{1}{m+n} \left( \sum s_i + \sum t_j \right)$
5: $\tilde{s} \leftarrow s - \alpha 1_m$
6: $\tilde{t} \leftarrow t - \alpha 1_n$

Apparently, the solution is given by
\[
\alpha_\ast := \frac{1^T_{m+n} z}{1^T_{m+n} 1_{m+n}} = \frac{1^T_{m+n} z}{m + n},
\]
and hence $\tilde{s}, \tilde{t}$ with
\[
\begin{bmatrix} \tilde{s} \\ \tilde{t} \end{bmatrix} := z - \alpha_\ast 1_{m+n}
\]
are normalized Cauchy points for $A$.

As described above, if $A = [a_{ij}]$ is a Cauchy matrix, then corresponding Cauchy points can be computed by solving the system (5)–(6). Since the matrix $U$ has rank $m+n-1$ (cf. Theorem 2.2), the points can be computed by solving any full-rank subsystem of (5)–(6) having $m+n-1$ rows. Due to the simple structure of $U$, the solution of this subsystem can be computed in $\mathcal{O}(m+n)$. One possible algorithm is shown in Algorithm 1. At the end of the algorithm we normalize the computed Cauchy points (according to Definition 2.3), which can be achieved in $\mathcal{O}(m+n)$ as well. Note that only the first row and column of $A$ are accessed by the algorithm.

If we do not know whether $A$ is a Cauchy matrix, we can still apply Algorithm 1 to $A$. Since the algorithm only considers the first row and column of $A$, it then costs (at most) $mn$ comparisons to check whether indeed $A = C(\tilde{s}, \tilde{t})$.

We summarize these observations in the following result.

Theorem 2.4. Let $A \in \mathbb{C}^{m,n}$ be given. If $A$ is a Cauchy matrix, then normalized Cauchy points $\tilde{s} \in \mathbb{C}^m$, $\tilde{t} \in \mathbb{C}^n$ with $A = C(\tilde{s}, \tilde{t})$ can be recovered using Algorithm 1 in $\mathcal{O}(m+n)$. Moreover, it can be decided in $\mathcal{O}(mn)$ whether $A$ is a Cauchy matrix.

Note that neither the recovery of Cauchy points, nor recognizing Cauchy structure can be achieved asymptotically faster than stated in the theorem.

3 Approximation with Cauchy matrices

In order to (best) approximate a given matrix $A \in \mathbb{C}^{m,n}$ by a Cauchy matrix we would ideally like to solve the nonlinear optimization problem (3). As described
in the Introduction, we will instead solve the linearization of this problem given by (4). Using the notation of Section 2, this standard linear least squares problem can be equivalently written as (cf. (5)–(6))

$$\min_{s,t} \|U [s t] - b\|_2^2.$$ (8)

Algorithm 1 from Section 2 is clearly inappropriate in this context, as there is no guarantee that the submatrix of $U$ picked for the reconstruction of the Cauchy points yields any useful global approximation of the given data when $A$ is not a Cauchy matrix. Our main goal in Section 3.1 is to derive an algorithm of optimal complexity $O(mn)$ for solving (8). In Section 3.2 we relate the (optimal) solution obtained by this algorithm to the original problem (3).

### 3.1 Fast solution of the least squares problem

We will solve the least squares problem (8) using the singular value decomposition of the matrix $U$. We have already characterized the kernel of $U$ in Theorem 2.2. In the following result we obtain a complete characterization the nonzero singular values and corresponding singular vectors.

**Theorem 3.1.** The nonzero singular values of the matrix $U$ in (6) are

- $\sqrt{m+n}$ (of multiplicity one),
- $\sqrt{m}$ (of multiplicity $n-1$),
- $\sqrt{n}$ (of multiplicity $m-1$).

Moreover, the corresponding right singular vectors can be characterized as

- $\sqrt{m+n}$: $\text{span}\left\{ \begin{bmatrix} n & m \\ m & -1 \end{bmatrix} \right\}$,
- $\sqrt{m}$: $\text{span}\left\{ \begin{bmatrix} 0 \\ m \\ v \end{bmatrix} : v \in \mathbb{C}^n, 1^T_n v = 0 \right\}$,
- $\sqrt{n}$: $\text{span}\left\{ \begin{bmatrix} v \\ 0_n \end{bmatrix} : v \in \mathbb{C}^m, 1^T_m v = 0 \right\}$.

and the corresponding left singular vectors can be characterized as

- $\sqrt{m+n}$: $\text{span}\{1_{mn}\}$,
- $\sqrt{m}$: $\text{span}\{1_m \otimes v : v \in \mathbb{C}^n, 1^T_n v = 0\}$,
- $\sqrt{n}$: $\text{span}\{v \otimes 1_n : v \in \mathbb{C}^m, 1^T_m v = 0\}$.

**Proof.** The claims can be verified by straightforward computations using the matrix

$$U^T U = \begin{bmatrix} nI_m & -1_{m,n} \\ -1_{n,m} & mI_n \end{bmatrix}.$$
for the right singular vectors, and the matrix
\[ UU^T = I_m \otimes 1_n 1_n^T + 1_m 1_m^T \otimes I_n \]
for the left singular vectors.

We next derive a matrix that can be used for constructing an orthogonal basis of the eigenspaces of \( U^T U \).

**Lemma 3.2.** Let \( m > 1 \), set \( n := m - 1 \) and \( \nu_j := \sqrt{1 + \frac{1}{j}} \) for \( j = 1, \ldots, n \). Then the unreduced upper Hessenberg matrix
\[
Q_m := \begin{bmatrix}
\nu_1 & 2\nu_2 & \cdots & n\nu_n \\
-\nu_1 & 2\nu_2 & \cdots & n\nu_n \\
-\nu_2 & \ddots & \ddots & \ddots \\
\vdots & \ddots & \ddots & \ddots \\
\nu_n & & & -\nu_n
\end{bmatrix}
\in \mathbb{R}^{m,m-1}
\tag{9}
\]
has orthonormal columns and satisfies \( 1_q^T Q_m = 0 \).

**Proof.** Let \( q_i, q_j \) be the \( i \)th and \( j \)th column of \( Q \), respectively, and assume without loss of generality that \( 1 \leq i < j \leq n \). We compute
\[
q_i^T q_j = \sum_{k=1}^{i} \frac{1}{j\nu_i\nu_j} - \frac{1}{j\nu_i\nu_j} = 1 + \frac{1}{j} = 1,
\]
For \( 1 \leq j \leq n \) we have
\[
q_j^T q_j = \sum_{k=1}^{j} \frac{1}{j^2\nu_j^2} + \frac{1}{j\nu_j} \frac{1}{j\nu_j} = \frac{1}{j\nu_j} - \frac{1}{j\nu_j} = 0,
\]
and finally \( 1_q^T q_j = \frac{1}{j\nu_j} - \frac{1}{j\nu_j} = 0 \). \( \square \)

The matrix \( Q_m \) in (9) yields an explicit matrix of right singular vectors of \( U \).

**Corollary 3.3.** The matrix \( V := [V_{m+n}, V_m, V_n, V_0] \in \mathbb{R}^{m+n,m+n} \) with
\[
V_{m+n} := (\frac{n^2}{m} + n) - \frac{1}{2} \begin{bmatrix} \frac{n}{m} \nu_{m} \\ -1_n \end{bmatrix}, \quad V_0 := (m + n) - \frac{1}{2} \begin{bmatrix} 1_{m+n} \\ \nu_{n} \end{bmatrix},
\]
\[
V_m := \begin{bmatrix} 0_{m,n-1} & Q_{m} \\ Q_{n} & 0_{n,m-1} \end{bmatrix}, \quad V_n := \begin{bmatrix} Q_{n} \\ 0_{n,m-1} \end{bmatrix},
\]
is orthogonal and consists of right singular vectors of \( U \) (where the indices indicate the corresponding squared singular values).
Using the matrix $V$ from the preceding corollary we get the diagonalization $U^T U = V \Lambda V^T$, where $\Lambda$ is the diagonal matrix containing the squared singular values of $U$. Thus we can solve the least squares problem (8) using $U^+$, the Moore-Penrose pseudoinverse of $U$, which gives

$$\begin{bmatrix}s \\ t\end{bmatrix} = U^+ b = V \Lambda^+ V^T U^T b, \quad b = \text{vec}(A^{[-T]}).$$

(10)

**Theorem 3.4.** The computation of $U^+ b$ in (10) can be carried out $O(mn)$.

**Proof.** Form the nonzero structure of $U$ it is easily seen that the product $U^T b$ can be computed with $O(mn)$ operations. Since multiplication with $\Lambda^+$ obviously takes $O(m + n)$ operations, we only need to study the multiplications with $V$ and $V^T$.

By the construction of $V$ it suffices to show that matrix-vector multiplications with $Q_m$ and $Q_m^T$ as defined in (9) can be carried out fast. Consider the decomposition $Q_m = W + H$, where $W$ captures the nonzeros in the upper triangle of $Q_m$, and $H$ the remaining subdiagonal. Then $Q_m x = W x + H x$ and $Q_m^T x = W^T x + H^T x$ for all $x \in \mathbb{C}^{m-1}$ and $x \in \mathbb{C}^m$, respectively. Clearly, the cost of evaluating the two products $H x$ and $H^T x$ is $O(m)$, so that it remains to consider the products $W x$ and $W^T x$.

For the first product we note that the $i$th row of $W$ is of the form

$$w_i = \begin{bmatrix} 0 & \ldots & 0 & i \nu_i & \ldots & n \nu_i \end{bmatrix}, \quad i = 1, \ldots, m$$

(there are $i - 1$ leading zeros). Thus,

$$w_i x = iv_i x_i + w_{i+1} x,$$

and hence the product $W x$ can be computed from bottom to top in $O(m)$.

For the second product we note that the $j$th column of $W$ is of the form

$$\begin{bmatrix} j \nu_j & \ldots & j \nu_j & 0 & \ldots & 0 \end{bmatrix}^T = j \nu_j \begin{bmatrix} 1_j \\ 0_{n-j} \end{bmatrix}, \quad j = 1, \ldots, m - 1.$$

From this it is easy to see that $W^T x$ can also be computed in $O(m)$, so that the total cost of computing $U^+ b$ remains at $O(mn)$. \qed

The overall algorithm is shown in Algorithm 2. Note that the points $s$ and $t$ obtained from $U^+ b$ are automatically normalized, and that the only operation that takes $mn$ operations is the evaluation of the product $U^T b$, which can be nicely expressed via

$$U^T b = \begin{bmatrix} A^{[-1]1_m} \\ -A^{[-T]1_n} \end{bmatrix}.$$  

Since all other operations are linear in $m + n$, Algorithm 2 is applicable in an online setting, where the data $A$ (and thus the entries of $b$) is obtained incrementally by rows and columns, without leaving the quadratic complexity regime.
Algorithm 2 Minimum 2-norm solution of the least squares problem (8).

**Input:** Matrix \( A \in \mathbb{C}^{m,n} \), \( a_{ij} \neq 0 \) for all \( i,j \); matrices \( V \) and \( \Lambda \) from the eigendecomposition of \( U^T U \).

**Output:** \( \hat{z} = U^+ \text{vec}(A[-1]) \) with minimum 2-norm over all solutions of (8).

1. \( x \leftarrow A[-1]_1^m \) \{The only \( O(mn) \) operation\}
2. \( z_1 \leftarrow V^T x \)
3. \( z_2 \leftarrow \Lambda + z_1 \) \{All three operations in \( O(m+n) \)\}
4. \( \hat{z} \leftarrow V z_2 \)

### 3.2 Approximation bounds

For each matrix \( A \in \mathbb{C}^{m,n} \) a minimum 2-norm solution \( \hat{z} = [\hat{s}, \hat{t}] \) of the least squares problem (8) and hence of (4) can be computed in \( O(mn) \) using Algorithm 2. Of course, without further assumptions we cannot expect that \( \hat{z} \) closely approximates the solution of the nonlinear problem (3). In fact, it is not even guaranteed that \( \hat{s}, \hat{t} \) are Cauchy points. Below we will derive a condition under which Algorithm 2 yields Cauchy points. Under this condition we will derive a bound on \( \|A - C(\hat{s}, \hat{t})\|_F \), and we will bound \( \|[\hat{s}] - [\hat{s}]\|_2 \) for a perturbed Cauchy matrix \( A = C(s, t) + N \).

In our results we will use the Hadamard (elementwise) product of matrices, i.e., \([a_{ij}] \odot [b_{ij}] := [a_{ij} b_{ij}]\), which is Frobenius norm submultiplicative, i.e. \( A \odot B\|_F \leq \|A\|_F \|B\|_F \); see, e.g., [5, equation (3.3.5)]. We will also use the matrix maximum norm \( \|A\|_M := \max_{i,j} |a_{ij}| \). We start with a simple lemma that will be used several times later.

**Lemma 3.5.** Let \( A = [a_{ij}] \in \mathbb{C}^{m,n} \) with \( \|A\|_F < 1 \) be given. Then for \( B = \begin{bmatrix} a_{ij} \overline{1+a_{ij}} \\ \overline{1-a_{ij}} \end{bmatrix} \) and \( B = \begin{bmatrix} a_{ij} \overline{1+a_{ij}} \\ \overline{1-a_{ij}} \end{bmatrix} \) we have

\[ \|B\|_F \leq \|A\|_F + O \left( \|A\|_F^2 \right). \]

**Proof.** Since \( \|A\|_F < 1 \) we have \( |a_{ij}| < 1 \) for all \( i,j \), so that

\[ \frac{a_{ij}}{1 \pm a_{ij}} = a_{ij} \left( 1 \mp a_{ij} + O \left( |a_{ij}|^2 \right) \right) = a_{ij} \mp a_{ij}^2 + O \left( |a_{ij}|^3 \right), \]

and hence

\[ \|A\|_F \leq \|A\|_F + \|A \odot A\|_F + O \left( \|a_{ij}\|^3 \right) \leq \|A\|_F + \|A\|_F^2 + O \left( \|A\|_F^3 \right), \]

which is even stronger than the statement we needed to prove. \( \square \)

The following results connect the residuals of (3) and (4). It shows that if for given \( \hat{s}, \hat{t} \) the relative residual of the linearization (4) is reasonably small, then \( \hat{s}, \hat{t} \) are Cauchy points, and their relative error with respect to the original problem (3) is small as well. Note that the theorem applies in particular to
the output of Algorithm 2, since it computes an optimal solution \( \hat{s}, \hat{t} \) for the linearization (4).

**Theorem 3.6.** Let \( A \in \mathbb{C}^{m,n} \), \( \hat{s} \in \mathbb{C}^{m} \), and \( \hat{t} \in \mathbb{C}^{n} \). Define the implied residual matrix corresponding to (4) by \( R := A^{-1} - D(\hat{s}, \hat{t}) \).

(1) If
\[
\| A \odot R \|_M =: \beta < 1,
\]
then
\[
\min_{i,j} |\hat{s}_i - \hat{t}_j| \geq \| A \|_M^{-1} (1 - \beta).
\]

In particular, \( \hat{s} \in \mathbb{C}^{m} \), \( \hat{t} \in \mathbb{C}^{n} \) are Cauchy points. Moreover,
\[
\frac{\| A - C(\hat{s}, \hat{t}) \|_F}{\| A \|_F} \leq \frac{\beta}{1 - \beta}.
\]

(2) If \( \| A \odot R \|_F < 1 \), then
\[
\frac{\| A - C(\hat{s}, \hat{t}) \|_F}{\| A \|_F} \leq \| A \odot R \|_F + O(\| A \odot R \|_F^2).
\]

Proof. (1) Let \( R = [r_{ij}] \), then for all \( i, j \) we get
\[
|\hat{s}_i - \hat{t}_j| = \left| \frac{1}{a_{ij}} - r_{ij} \right| = \frac{|1 - a_{ij} r_{ij}|}{|a_{ij}|} \geq \| A \|_M^{-1} (1 - \beta),
\]
which shows the lower bound on \( \min_{i,j} |\hat{s}_i - \hat{t}_j| \).

For the second inequality we compute
\[
a_{ij} - \frac{1}{\hat{s}_i - \hat{t}_j} = a_{ij} - \frac{1}{a_{ij} - r_{ij}} = a_{ij} \left( 1 - \frac{1}{1 - a_{ij} r_{ij}} \right) = a_{ij} \frac{a_{ij} r_{ij}}{1 - a_{ij} r_{ij}},
\]
so that
\[
\left| a_{ij} - \frac{1}{\hat{s}_i - \hat{t}_j} \right| \leq \frac{\beta}{1 - \beta},
\]
giving \( \| A - C(\hat{s}, \hat{t}) \|_F \leq \frac{\beta}{1 - \beta} \| A \|_F \).

(2) If \( \| A \odot R \|_F < 1 \), then obviously \( \| A \odot R \|_M < 1 \), so that \( \hat{s}, \hat{t} \) are Cauchy points. We use
\[
a_{ij} - \frac{1}{\hat{s}_i - \hat{t}_j} = a_{ij} \frac{a_{ij} r_{ij}}{1 - a_{ij} r_{ij}},
\]
which implies \( \| A - C(\hat{s}, \hat{t}) \|_F \leq \| A \odot B \|_F \) with \( B = \left[ \frac{a_{ij} r_{ij}}{1 - a_{ij} r_{ij}} \right] \). The inequality now follows from the Frobenius norm submultiplicativity of the Hadamard product and Lemma 3.5.
The condition (11) can be written as

$$\max_{i,j} \left| \frac{(\hat{s}_i - \hat{t}_j) - a_{ij}^{-1}}{a_{ij}^{-1}} \right| = \beta < 1.$$  

In words, the maximal relative error in the linear equations (2) that is made by the vectors \(\hat{s}, \hat{t}\) has to be smaller than one. This appears to be a natural and in fact minimal assumption on the output of Algorithm 2 so that it gives any useful information about the optimization problems (3) and (4).

In the next result we investigate how closely the output of Algorithm 2 approximates the Cauchy points defining a perturbed Cauchy matrix \(A\).

**Theorem 3.7.** Let \(A = C(\hat{s}, \hat{t}) + N \in \mathbb{C}^{m,n}\), where \(\hat{s}, \hat{t}\) are normalized Cauchy points, and let \(\tilde{s} \in \mathbb{C}^m, \tilde{t} \in \mathbb{C}^n\) be a minimum 2-norm solution of the least squares problem (8), i.e., the output of Algorithm 2 applied to \(A\). If

$$\|D(\tilde{s}, \tilde{t}) \odot N\|_F < 1,$$  

then

$$\frac{\|\left[\begin{array}{c} \tilde{s} \\ \tilde{t} \end{array}\right] - \left[\begin{array}{c} s \\ t \end{array}\right]\|_2}{\|\left[\begin{array}{c} s \\ t \end{array}\right]\|_2} \leq \frac{2}{\min\{\sqrt{m}, \sqrt{n}\}} \left(\|D(\tilde{s}, \tilde{t}) \odot N\|_F + \mathcal{O}\left(\|D(\tilde{s}, \tilde{t}) \odot N\|_F^2\right)\right).$$

**Proof.** Let us denote \(C = C(\hat{s}, \hat{t}), N = [n_{ij}]\) and define

\[ B = [b_{ij}] := A^{[-T]} - C^{[-T]}, \quad \xi_{\max} := \max_{i,j} |\tilde{s}_i - \tilde{t}_j|. \]

Then for all \(i, j\) we get

$$|b_{ji}| = \left| \frac{1}{\tilde{s}_i - \tilde{t}_j} + n_{ij} \right| - \left| \frac{1}{\hat{s}_i - \hat{t}_j} \right| = \frac{1}{\tilde{s}_i - \tilde{t}_j} \frac{\tilde{s}_i - \tilde{t}_j}{1 + (\tilde{s}_i - \tilde{t}_j)n_{ij}} \leq \xi_{\max} \left| \frac{(\tilde{s}_i - \tilde{t}_j)n_{ij}}{1 + (\tilde{s}_i - \tilde{t}_j)n_{ij}} \right|. \tag{14}$$

Using Lemma 3.5 we obtain

$$\|B\|_F \leq \xi_{\max} \left\| \frac{(\tilde{s}_i - \tilde{t}_j)n_{ij}}{1 + (\tilde{s}_i - \tilde{t}_j)n_{ij}} \right\|_F \leq \xi_{\max} \left( \|D \odot N\|_F + \mathcal{O}\left(\|D \odot N\|_F^2\right)\right).$$

Recall from Section 2 (cf. (10)) that

\[ \left[\begin{array}{c} \tilde{s} \\ \tilde{t} \end{array}\right] = U^+ \text{vec}(A^{[-T]}) = U^+ (\text{vec}(C^{[-T]}) + \text{vec}(B)) = \left[\begin{array}{c} \tilde{s} \\ \tilde{t} \end{array}\right] + U^+ \text{vec}(B). \]

We thus get

$$\|\left[\begin{array}{c} \tilde{s} \\ \tilde{t} \end{array}\right] - \left[\begin{array}{c} s \\ t \end{array}\right]\|_2 = \|U^+ \text{vec}(B)\|_2 = \|U^+\|_2 \|B\|_F \leq \frac{1}{\min\{\sqrt{m}, \sqrt{n}\}} \xi_{\max} \left( \|D \odot N\|_F + \mathcal{O}\left(\|D \odot N\|_F^2\right)\right),$$
where we used Theorem 3.1. Finally, the bound follows from
\[ \xi_{\max} \leq \max_{i,j} \{|\tilde{s}_i| + |\tilde{t}_j|\} \leq 2 \left\| \begin{bmatrix} \tilde{s} \\ \tilde{t} \end{bmatrix} \right\|_{\infty} \leq 2 \left\| \begin{bmatrix} \tilde{s} \\ \tilde{t} \end{bmatrix} \right\|_2. \]

The condition (13), i.e.,
\[ \left\| \left( \begin{bmatrix} \tilde{s} \\ \tilde{t} \end{bmatrix} - \begin{bmatrix} \hat{s} \\ \hat{t} \end{bmatrix} \right)_{n_{ij}} \right\|_F < 1, \]
ensures that the small values of the Cauchy matrix \( C(\tilde{s}, \tilde{t}) \) are not polluted with large values of noise. Note also that \( \|D \odot N\|_F \leq \xi_{\max} \|N\|_F \), so that the bound in the theorem can also be phrased in terms of \( \xi_{\max} \|N\|_F \), provided this value is smaller than one.

A more compact, direct bound for the relative Cauchy point recovery error can be obtained if \( \|D \odot N\|_M \) is bounded away from one.

**Theorem 3.8.** In the notation of Theorem 3.7, if
\[ \|D \odot N\|_M = \gamma < 1, \]
then
\[ \frac{\left\| \begin{bmatrix} \tilde{s} \\ \tilde{t} \end{bmatrix} - \begin{bmatrix} \hat{s} \\ \hat{t} \end{bmatrix} \right\|_2}{\left\| \begin{bmatrix} \tilde{s} \\ \tilde{t} \end{bmatrix} \right\|_2} \leq \frac{\sqrt{n} + \sqrt{m}}{\min\{\sqrt{m}, \sqrt{n}\}} \frac{\gamma}{1 - \gamma}. \]

**Proof.** The bound can be obtained by the same technique as in Theorem 3.7, where we bound (14) using \( \gamma \), viz.
\[ |b_{ji}| \leq |\tilde{s}_i - \tilde{t}_j| \frac{\gamma}{1 - \gamma}, \]
and noting that
\[ \left\| D(\tilde{s}, \tilde{t}) \right\|_F = \left\| \tilde{s} \otimes 1^T_n - 1_m \otimes \tilde{t}^T \right\|_F \leq \left\| \tilde{s} \otimes 1^T_n \right\|_F + \left\| 1_m \otimes \tilde{t}^T \right\|_F \]
\[ \leq \sqrt{n} \left\| \tilde{s} \right\|_2 + \sqrt{m} \left\| \tilde{t} \right\|_2 \leq (\sqrt{m} + \sqrt{n}) \left\| \begin{bmatrix} \tilde{s} \\ \tilde{t} \end{bmatrix} \right\|_2, \]
so that
\[ \|B\|_F \leq \left\| D(\tilde{s}, \tilde{t}) \right\|_F \frac{\gamma}{1 - \gamma} \leq (\sqrt{m} + \sqrt{n}) \left\| \begin{bmatrix} \tilde{s} \\ \tilde{t} \end{bmatrix} \right\|_2 \frac{\gamma}{1 - \gamma}. \]

Proceeding exactly as in Theorem 3.7 yields the desired bound. \( \square \)

Note that the constant on the right hand side of (15) reduces to 2 when \( m = n \).

We will now illustrate the theoretical results obtained in this section by a numerical example. We generated a vector \( s \in \mathbb{C}^{100} \), where the real part is drawn from the standard normal distribution, i.e. \( \Re(s_i) \sim \mathcal{N}(0,1) \), and with imaginary parts set to \( i \), i.e. \( \Im(s) = i1_m \). The vector \( t \in \mathbb{C}^{200} \) has its real part
Figure 1: Relative data approximation error corresponding to the Cauchy points obtained by Algorithm 2 (blue), the true Cauchy points (black), and the bound (12) (red). The black and blue lines are visually almost indistinguishable.

drawn at random from $N(0, 1)$ as well, but the imaginary part is set to $-i$. We denote by $\tilde{s}$ and $\tilde{t}$ the corresponding normalized Cauchy points.

The particular choice for the imaginary parts of $s$ and $t$ ensures that
$$\frac{\min_{i,j} |s_i - t_j|}{\|C\|_F} \in \mathcal{O}(1),$$

so that $C(\tilde{s}, \tilde{t})$ can be perturbed by large levels of noise without violating the condition that the entries of the perturbed matrix must be nonzero. Note that if the ratio of the largest and smallest distance is large, the computations with the reciprocals of the matrix' entries could induce large numerical cancellations (see Section 4).

For each value $\delta$ of a series of target noise levels in $[0, 1]$ we draw a matrix $N$ with entries from $N(0, 1)$ and set
$$C_\delta = C(\tilde{s}, \tilde{t}) + \delta \frac{\|C(\tilde{s}, \tilde{t})\|_F}{\|N\|_F} N,$$

so that $\delta$ is the relative norm of the perturbation term with respect to $C(\tilde{s}, \tilde{t})$. We then use Algorithm 2 to recover Cauchy points $\hat{s}, \hat{t}$ that so that the Cauchy matrix $C(\hat{s}, \hat{t})$ approximates $C_\delta$.

Figure 1 shows, for each noise level $\delta$, the relative approximation error $\frac{\|C_\delta - C(\hat{s}, \hat{t})\|_F}{\|C_\delta\|_F}$. For comparison purpose, we also show the relative error that
is achieved by the original Cauchy points $\tilde{s}$ and $\tilde{t}$, and the bound (12). The data shows that the Cauchy points recovered by Algorithm 2 allow for approximation of the given data matrix $C_\delta$ with a Cauchy matrix with approximation error linear in the noise level, and that these points are on par with the original Cauchy points.

The recovered Cauchy points are, however, different from the original ones. Figure 2 shows the relative recovery error $\frac{\|\tilde{z} - \hat{z}\|_2}{\|\tilde{z}\|_2}$, where we have set $\tilde{z} := [\tilde{s} \, \tilde{t}]$ and $\hat{z} := [\hat{s} \, \hat{t}]$. As for the data approximation error, the recovery error behaves linearly in the noise level.

4 Concluding remarks

We presented an efficient algorithm for the approximation of a given matrix with a Cauchy matrix. Our approach for solving the approximation problem is based on the efficient solution of a simple linear least squares problem, and the key tool for its solution is the explicit construction of the pseudoinverse of a peculiarly structured matrix. Hence our algorithm obtains its optimal, minimum norm solution by orthogonal transformations of the original data, which is favorably from a numerical point of view. We have not (yet) conducted a formal error analysis of our algorithms, however. Besides this error analysis, it would be very interesting to investigate whether similar algorithms can be devised for the approximation of a given matrix with displacement structured matrices of
displacement rank greater than one.

Acknowledgements The work of R. Luce is supported by Deutsche Forschungsgemeinschaft, cluster of excellence “UniCat”.

References

[1] I. Gohberg, T. Kailath, and V. Olshevsky. Fast Gaussian elimination with partial pivoting for matrices with displacement structure. Math. Comp., 64(212):1557–1576, 1995.

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

[3] Gene H. Golub and Charles F. Van Loan. Matrix computations. Johns Hopkins Studies in the Mathematical Sciences. Johns Hopkins University Press, Baltimore, MD, fourth edition, 2013.

[4] L. Greengard and V. Rokhlin. A fast algorithm for particle simulations. J. Comput. Phys., 73(2):325–348, 1987.

[5] Roger A. Horn. The Hadamard product. In Charles R. Johnson, editor, Matrix theory and applications (Phoenix, AZ, 1989), volume 40 of Proc. Sympos. Appl. Math., pages 87–169. Amer. Math. Soc., Providence, RI, 1990.

[6] Victor Y. Pan. Structured matrices and polynomials. Birkhäuser Boston, Inc., Boston, MA; Springer-Verlag, New York, 2001. Unified superfast algorithms.