Computing accurate eigenvalues and eigenvectors of symmetric quasi-Cauchy matrices

Froilán M. Dopico

Dpto. de Matemáticas. Universidad Carlos III de Madrid. Spain

Setting the Problem

Definition. Quasi-Cauchy matrix:

$$A = D_1 C D_2 \quad D_1, D_2 \quad \text{diagonal matrices}$$

and C is a Cauchy matrix: $C_{ij} = \frac{1}{x_i + y_j} \quad i, j = 1 : n$, where x_i and y_j are given floating point numbers.

Problem: Usual algorithms (QR, Divide and Conquer, Jacobi, J-Orthogonal Jacobi [9],...) **do not compute** accurate eigenvalues and eigenvectors of REAL SYMMETRIC quasi-Cauchy matrices.

Goal: Compute EIGENVALUES and EIGENVECTORS of REAL SYMMETRIC ($D_1 = D_2, x_i = y_i$) QUASI-CAUCHY matrices with HIGH RELATIVE ACCURACY and in $O(n^3)$ flops.

Previous work: Demmel [2] gave a $O(n^3)$ algorithm to compute **SVD** of quasi-Cauchy matrices with high relative accuracy.

Numerical Experiment 1

Consider the 100×100 Cauchy matrix

$$C_{ij} = \frac{1}{x_i + x_j} \quad , \quad x_i = (-1)^{(i-1)} + 2^{-40} * (i-1),$$

with $\kappa(C) = 7.8 * 10^{73}$. Eigenvalues and eigenvectors using **Mathematica with 120-decimal digit precision** are compared with algorithms in MATLAB ($\epsilon \approx 1.1 * 10^{-16}$).

METHOD	$\max_i \frac{\lvert\lambda_i - \widehat{\lambda}_i\rvert}{\lvert\lambda_i\rvert}$	$\max_i \lVert v_i - \widehat{v}_i \rVert$
MATLAB (eig)	$1.1 * 10^{55}$	1.41
Stand. J-Orthog.	$9.7 * 10^{54}$	1.41

- #(rel. error $\lambda > 1$) = 64. (Stand. J-Orthog. and MATLAB).

Numerical Experiment 2

Consider the 100×100 Cauchy matrix

$$x_i = (i - 0.5) \quad i = 1 : (n-1) \quad ; \quad x_n = -(n - 0.5)$$

with $\kappa(C) = 3.53 * 10^{147}$. **Mathematica computations with 200-decimal digit precision.**

METHOD	$\max_i \frac{\lvert\lambda_i - \widehat{\lambda}_i\rvert}{\lvert\lambda_i\rvert}$	$\max_i \lVert v_i - \widehat{v}_i \rVert$
MATLAB (eig)	$2.6 * 10^{131}$	1.41
Stand. J-Orthog.	$4.4 * 10^{130}$	1.41

- #(rel. error $\lambda > 1$) = 77 (Stand. J-Orthog.) , 78 (MATLAB).

Previous work on General matrices (1)

The following algorithm was introduced by Demmel, Gu, Eisenstat, Slapničar, Veselić, Drmač [3] to compute the SVD of general matrices with high relative accuracy:

`Algorithm to compute SVD of general` A.

STEP 1: Compute a **rank revealing decomposition (RRD)** $A = XDY^T$, where D is a nonsingular diagonal matrix and X, Y are well conditioned matrices.

STEP 2: Apply a Jacobi-type algorithm of Demmel et al. to the computed RRD to compute the SVD of $A = U\Sigma V^T$.

STEP 2 (Algorithm 3.1 in [3]): SVD of $A = XDY^T$.

1. QR factorization with column pivoting of XD, $XD = QRP$. Thus $A = QRPY^T$.
2. Multiply to get $W = RPY^T$. Thus $A = QW$.
3. SVD of W with one-sided Jacobi; $W = \overline{U}\Sigma V^T$. Thus $A = Q\overline{U}\Sigma V^T$.
4. Multiply $U = Q\overline{U}$. Thus $A = U\Sigma V^T$

Previous work on General matrices (2)

Next Theorem was proved in [3] for the Algorithm in page 3.

Notation: $R' \equiv D'^{-1}R$, where D' is a diagonal matrix chosen so that R' is as well conditioned as possible (usually $\kappa(R') \approx 1$, and at worst $\kappa(R') = O(n^{3/2}\kappa(X))$). Letters with a hat denote computed magnitudes, and without a hat exact magnitudes. ϵ is machine precision.

Theorem: Let us assume *that the computed RRD satisfies*

$$\|X - \widehat{X}\|_2 = O(\epsilon)\|X\|_2 \quad , \quad \|Y - \widehat{Y}\|_2 = O(\epsilon)\|Y\|_2$$

$$|D_{ii} - \widehat{D}_{ii}| = O(\epsilon)|D_{ii}| \quad \text{and} \quad \widehat{D} \quad \text{diagonal.}$$

Then

$$|\sigma_i - \widehat{\sigma}_i| = O(\epsilon\kappa(R')\max(\kappa(X),\kappa(Y)))\sigma_i,$$

$$\theta(u_i, \widehat{u}_i) \text{ or } \theta(v_i, \widehat{v}_i) = \frac{O(\epsilon\kappa(R')\max(\kappa(X),\kappa(Y)))}{relgap_i}$$

where $\quad relgap_i = \min\left(\min_{j\neq i} \frac{|\sigma_i - \sigma_j|}{\sigma_i}, 2\right).$

Conclusion: Computing an accurate enough RRD of A is sufficient to compute a SVD with high relative accuracy.

Previous work on quasi-Cauchy matrices

Gaussian elimination with complete pivoting **(GECP)** is used by Demmel [2] to compute an accurate RRD, $\mathbf{A} = \mathbf{XDY^T}$, but **not in the usual way**.

Notice that after having done k steps of GE on $A \in \mathbb{R}^{n \times n}$.

$$A = \begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix} = \begin{bmatrix} L_{11} & 0 \\ A_{21}U_{11}^{-1} & I \end{bmatrix} \begin{bmatrix} U_{11} & L_{11}^{-1}A_{12} \\ 0 & S^{(k)} \end{bmatrix}$$

where $A_{11} = L_{11}U_{11}$ and the **k-th Schur Complement**

$$S^{(k)} = A_{22} - A_{21}A_{11}^{-1}A_{12} \in \mathbb{R}^{(n-k)\times(n-k)}.$$

- $D_{k+1,k+1} = S_{11}^{(k)}$.
- $X(k+1:n, k+1) = S^{(k)}(:,1)/S_{11}^{(k)}$.
- $Y^T(k+1, k+1:n) = S^{(k)}(1,:)/S_{11}^{(k)}$.
- The pivoting strategy can be introduced in this scheme.

For **quasi-Cauchy** matrices:

$$S_{rs}^{(k)} = S_{rs}^{(k-1)} \frac{(x_r - x_k)(y_s - y_k)}{(x_k + y_s)(x_r + y_k)}.$$

The elements of all the Schur complements are computed with relative accuracy $O(\epsilon)$. Then, the RRD is computed with the required accuracy and cost $\frac{4}{3}n^3$ or $\frac{8}{3}n^3$ flops.

Computing eigenvalues and eigenvectors

Consider **SYMMETRIC quasi-Cauchy** real matrices

$$A = DCD \quad \text{where} \quad D \quad \text{diagonal matrix}$$

and C is a symmetric Cauchy matrix:

$$C_{ij} = \frac{1}{x_i + x_j} \quad i, j = 1 : n.$$

First Method: Signed SVD

(Dopico, Molera, Moro [4])

If a SVD with high relative accuracy can be computed this method produces eigenvalues and eigenvectors with high relative accuracy.

Outline of the method:

1. Compute SVD of quasi-Cauchy matrix, $A = U\Sigma V^T$, using Demmel's algorithm.
2. Assign the correct signs to the singular values

$$\lambda_i = (v_i^T u_i)\sigma_i, \quad (v_i^T u_i) = \pm 1$$

3. Eigenvectors $q_i = v_i$.

Method works when tight clusters of singular values are present with a convenient modification.

Numerical Experiment 1

METHOD	$\max_i \frac{\lvert\lambda_i - \widehat{\lambda}_i\rvert}{\lvert\lambda_i\rvert}$	$\max_i \lVert v_i - \widehat{v}_i \rVert$
MATLAB (eig)	$1.1 * 10^{55}$	1.41
Stand. J-Orthog.	$9.7 * 10^{54}$	1.41
Signed SVD	$2.8 * 10^{-15}$	$1.7 * 10^{-14}$

- #(rel. error $\lambda > 1$) = 64. (Stand. J-Orthog. and MATLAB).

Numerical Experiment 2

Consider the 100×100 Cauchy matrix

$$x_i = (i - 0.5) \quad i = 1 : (n-1) \quad ; \quad x_n = -(n - 0.5)$$

with $\kappa(C) = 3.53 * 10^{147}$. **Mathematica computations with 200-decimal digit precision.**

METHOD	$\max_i \frac{\lvert\lambda_i - \widehat{\lambda}_i\rvert}{\lvert\lambda_i\rvert}$	$\max_i \lVert v_i - \widehat{v}_i \rVert$
MATLAB (eig)	$2.6 * 10^{131}$	1.41
Stand. J-Orthog.	$4.4 * 10^{130}$	1.41
Signed SVD	$6.5 * 10^{-15}$	$3 * 10^{-14}$

- #(rel. error $\lambda > 1$) = 77 (Stand. J-Orthog.) , 78 (MATLAB).

Second (Symmetric) Method: J-Orthogonal

Remark: GECP and **STEP 2** in page 3 on a symmetric quasi-Cauchy matrix does not take advantage of symmetry ($A = XDY^T$ with $X \neq Y$).

Question: *Can we develop a symmetric algorithm?*

Answer: **Yes.**

We use a two step algorithm like the J-Orthogonal proposed by Veselić [9] and analyzed by Slapničar [7],

BUT WE NEED TO MODIFY THE FACTORIZATION STEP TO GET HIGH RELATIVE ACCURACY FOR QUASI-CAUCHY MATRICES.

1. **Factorization Step:** Compute an accurate enough *symmetric indefinite decomposition* (SID):

$$A = GJG^T \quad , \quad J = I_p \oplus -I_{r-p}.$$

2. Apply to this SID the implicit one-sided **J-orthogonal Jacobi method** of Veselić and Slapničar [9], [7] to obtain eigenvalues and eigenvectors with high relative accuracy.

Accurate $A = GJG^T$ (SID) of quasi-Cauchy

We only describe the **first step** of the computation using Bunch and Parlett's pivoting strategy [1]:

```
μ0 = max_ij |a_ij| , μ1 = max_i |a_ii| , α ≈ 0.64
if μ1 ≥ αμ0
    1 × 1 pivot s.t. |e11| = μ1.
else
    2 × 2 pivot s.t. |e21| = μ0.
end
```

where E is the $s \times s$ chosen pivot ($s = 1, 2$), and Π a permutation

$$\Pi A \Pi^T = \begin{bmatrix} E & C^T \\ C & B \end{bmatrix}.$$

The first step of the *diagonal pivoting method* is

$$\Pi A \Pi^T = \begin{bmatrix} I_s & 0 \\ CE^{-1} & I \end{bmatrix} \begin{bmatrix} E & 0 \\ 0 & S^{(s)} \end{bmatrix} \begin{bmatrix} I_s & E^{-1}C^T \\ 0 & I \end{bmatrix},$$

with the Schur complement

$$S^{(s)} = B - CE^{-1}C^T.$$

1×1 pivots as usual; for 2×2 do the spectral decomposition: [8]:

$$E = Q_s \Lambda_s Q_s^T = (Q_s \sqrt{|\Lambda_s|}) \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix} (\sqrt{|\Lambda_s|} Q_s^T).$$

Thus, $\Pi A \Pi^T = G^{(s)} J^{(s)} (G^{(s)})^T$ with

$$G^{(s)} = \begin{bmatrix} Q_s\sqrt{|\Lambda_s|} & 0 \\ CQ_s(\sqrt{|\Lambda_s|})^{-1} J_1 & I \end{bmatrix}, \; J^{(s)} = \begin{bmatrix} J_1 & 0 \\ 0 & S^{(s)} \end{bmatrix},$$

and $J_1 = diag(1, -1)$.

We compute the elements of all the Schur complements with relative error $O(\epsilon)$: if the pivot is 1×1 just as in Demmel's algorithm (pag. 5), and for **$\mathbf{2 \times 2}$ pivots** using

$$S_{rs}^{(k+1)} = S_{rs}^{(k-1)} \frac{(x_r - x_k)(x_s - x_k)}{(x_k + x_s)(x_r + x_k)} \frac{(x_r - x_{k+1})(x_s - x_{k+1})}{(x_{k+1} + x_s)(x_r + x_{k+1})}$$

Rounding errors in $A = GJG^T$

The following theorem is proved by using

- the 2×2 pivots fulfill $\kappa_2(E) \leq 4.6$ and $|(CE^{-1})_{ij}| \leq 2.78$.

Theorem: *Let $\widehat{G}\widehat{J}\widehat{G}^T$ be the SID computed by the previous algorithm applied to a symmetric quasi-Cauchy matrix A with precision ϵ.* **If the same sequence of pivots (in size and positions)** *is applied in* **exact** *arithmetic, a SID of A is obtained:*

$$A = GJG^T, \qquad \textit{such that}$$

1. $\widehat{J} = J$, *whenever* $100n\epsilon < 1/2$.
2. $\|\widehat{G}(:,j) - G(:,j)\|_2 = O(\epsilon)\|G(:,j)\|_2 \;\; \forall j$.
3. $\|\widehat{G}D - GD\|_2 = O(\epsilon)\|GD\|_2 \;\; \forall$ *diagonal nonsing.* D.
4. $\widehat{G}J\widehat{G}^T = (I+E)A(I+E)^T$, $\|E\|_2 = O(\epsilon) \min_D \kappa_2(GD)$.

Overall error in $A = GJG^T$ + J-Jacobi

Combining the previous result with **the error analysis done by Slapničar [7] for the implicit one-sided J-orthogonal Jacobi method**, we obtain that the computed eigenvalues and eigenvectors are the exact eigenvalues and eigenvectors of a *small multiplicative perturbation* of A:

$$(I+F)A(I+F)^T,$$

with

$$\|F\|_2 = O(\epsilon\,\gamma\,\kappa_2(GC_0)) \quad \text{and} \quad \gamma = \max_{1\leq i\leq M} \frac{\kappa_2(G_iC_i)}{\kappa_2(GC_0)},$$

where C_i are diagonal matrices such that

$$\|(G_iC_i)(:,j)\|_2 = 1 \quad \forall j.$$

for all the J-Orthogonal Jacobi iterates G_i. Numerical experiments show that $\gamma \approx 1$, but it is still an open problem to prove it.

Well-known results of *multiplicative perturbation theory* (Eisenstat, Ipsen [5], Li [6]) yield **high relative accuracy** for e-values and e-vectors:

$$|\widehat{\lambda}_i - \lambda_i| = O(\epsilon\gamma\kappa_2(GC_0))\ |\lambda_i|,$$

$$\theta(q_i, \widehat{q}_i) = \frac{O(\epsilon\gamma\kappa_2(GC_0))}{\min_{j\neq i} \frac{|\lambda_j-\lambda_i|}{|\lambda_i|}}.$$

Numerical Experiments (1)

Consider the 100×100 Cauchy matrix

$$C_{ij} = \frac{1}{x_i + x_j} \quad , \quad x_i = (-1)^{(i-1)} + 2^{-40} * (i-1),$$

with $\kappa(C) = 7.8 * 10^{73}$. Eigenvalues and eigenvectors using **Mathematica with 120-decimal digit precision** are compared with algorithms in MATLAB ($\epsilon \approx 1.1 * 10^{-16}$).

METHOD	$\max_i \frac{\vert\lambda_i - \widehat{\lambda}_i\vert}{\vert\lambda_i\vert}$	$\max_i \Vert v_i - \widehat{v}_i \Vert$
MATLAB (eig)	$1.1 * 10^{55}$	1.41
Stand. J-Orthog.	$9.7 * 10^{54}$	1.41
Signed SVD	$2.8 * 10^{-15}$	$1.7 * 10^{-14}$
Cauchy J-Orthog.	$1.1 * 10^{-14}$	$1.7 * 10^{-14}$

- #(rel. error $\lambda > 1$) = 64. (Stand. J-Orthog. and MATLAB).
- 1400 random symm. quasi-Cauchy matrices of 7 different types and $10 \leq n \leq 100$ also produce very satisfactory results.
- # Jacobi sweeps.

METHOD	MAX	MEAN	MIN
Signed SVD	8	5.3	2
Cauchy J-Orthog.	10	6.9	3

References

[1] J.R. Bunch and B. Parlett, *Direct methods for solving symmetric indefinite systems of linear equations*, SIAM J. Numer. Anal., 8 (1971), pp. 639-655.

[2] J. Demmel, *Accurate SVD of structured matrices*, SIAM J. Matrix Anal. Appl., 21 (1999), pp. 562-580.

[3] J. Demmel, M. Gu, S. Eisenstat, I. Slapničar, K. Veselić and Z. Drmač, *Computing the singular value decomposition with high relative accuracy*, Linear Algebra Appl. 299 (1999), pp. 21-80.

[4] F. M. Dopico, J. M. Molera and J. Moro, *An orthogonal high relative accuracy algorithm for the symmetric eigenproblem*, submitted.

[5] S. Eisenstat and I. Ipsen, *Relative perturbation techniques for singular value problems*, SIAM J. Numer. Anal., 32 (1995), pp. 1972-88.

[6] R.-C. Li, *Relative Perturbation Theory: (II) Eigenspace and Singular Subspace Variations*, SIAM J. Matrix Anal. Appl., 20 (1999), pp. 471-492.

[7] I. Slapničar, *Accurate Symmetric Eigenreduction by a Jacobi Method*, PhD Thesis, Fachbereich Mathematik Fernuniversität, Gesamthochschule Hagen, Germany, 1992.

[8] I. Slapničar, *Componentwise analysis of direct factorization of real symmetric and Hermitian matrices*, Linear Algebra Appl., 272 (1998), pp. 227-275.

[9] K. Veselić, *A Jacobi eigenreduction algorithm for definite matrix pairs*, Numer. Math., 64 (1993), pp. 241-269.

Numerical Experiments (2)

- 1400 random symm. quasi-Cauchy matrices of 7 different types and dimensions $10 \le n \le 100$.
- Use $C_{ij} = 1/(x_i + x_j) \Rightarrow$

$$C^{-1} = D'CD' \quad \text{with} \quad D'_{ii} = \frac{\prod_k (x_k + x_i)}{\prod_{k \ne i}(x_i - x_k)}.$$

- Compare with MATLAB:

$$\max \max_i \frac{|\lambda_i - \lambda_i^M|}{\epsilon ||A||_2}$$

Signed SVD = 101	J-Orthog. = 264

- The same for the inverse ($\lambda^{inv} = 1/\lambda$):

$$\max \max_i \frac{|\lambda_i^{inv} - \lambda_i^M(A^{-1})|}{\epsilon ||A^{-1}||_2}$$

Signed SVD = 58	J-Orthog. = 313

- Compare with the same algorithm for the inverse:

$$\max \max_i \frac{|\lambda_i - (\lambda_i(A^{-1}))^{-1}|}{\epsilon |\lambda_i|}$$

Signed SVD = 244	J-Orthog. = 932

- Number of matrices with rel. error > 1 in some eigenvalue

MATLAB = 907	Stand. J-Orthog. = 475

Numerical Experiment I

Consider the 100×100 Cauchy matrix

$$C_{ij} = \frac{1}{x_i + x_j} \quad , \quad x_i = (-1)^{(i-1)} + 2^{-40} * (i-1),$$

with $\kappa(C) = 7.8 * 10^{73}$. We compute spectral decomposition using **Mathematica with 120-decimal digit of precision** and compare with algorithms implemented in MATLAB ($\epsilon \approx 1.1 * 10^{-16}$).

METHOD	$\max_i \frac{\lvert\lambda_i - \widehat{\lambda}_i\rvert}{\lvert\lambda_i\rvert}$	$\max_i \lVert v_i - \widehat{v}_i \rVert$
MATLAB (eig)	$1.1 * 10^{55}$	1.41
Stand. J-Orthog.	$9.7 * 10^{54}$	1.41
Signed SVD	$2.8 * 10^{-15}$	$1.7 * 10^{-14}$
Cauchy J-Orthog.	$1.1 * 10^{-14}$	$1.7 * 10^{-14}$

Other interesting data:

- $1.9 * 10^{-62} \leq \lvert\lambda\rvert \leq 1.6 * 10^{12}$.
- $\min_i relgap_i(\lambda) = 0.62$; $\min_i relgap_i(\sigma) = 0$!!!
- #(rel. error $\lambda > 1$) = 64. (Stand. J-Orthog. and MATLAB).
- $\max(\kappa(X), \kappa(Y)) = 38.7$ (GECP).
- $\kappa(GD_G) = 30.5$ (SID).

Numerical Experiment II (1)

- 1400 random symm. quasi-Cauchy matrices of 7 different types and dimensions $10 \le n \le 100$.
- Use $C_{ij} = 1/(x_i + x_j) \Rightarrow$

$$C^{-1} = D'CD' \quad \text{with} \quad D'_{ii} = \frac{\prod_k (x_k + x_i)}{\prod_{k \neq i}(x_i - x_k)}.$$

- Compare with MATLAB: $\max_i \frac{|\lambda_i - \lambda_i^M|}{\epsilon ||A||_2}$

Signed SVD = 101	J-Orthog. = 264

- The same for the inverse ($\lambda_{inv} = 1/\lambda$):

Signed SVD = 58	J-Orthog. = 313

- Compare with the same algorithm for the inverse:

$$\max_i \frac{|\lambda_i - (\lambda_i^{(-1)})^{-1}|}{\epsilon |\lambda_i|}$$

Signed SVD = 244	J-Orthog. = 932

- Compare with MATLAB:
$\max_i \|v_i - v_i^M\| absgap_i(\sigma \text{ or } \lambda)/\epsilon$

Signed SVD = 47	J-Orthog. = 85

- The same for the inverse:

Signed SVD = 280	J-Orthog. = 370

Numerical Experiment II (2)

- Compare with the same algorithm for the inverse:

$$\max_i \|v_i - v_i(A^{-1})\| relgap_i(\sigma \text{ or } \lambda)/\epsilon$$

Signed SVD = 321	J-Orthog. = 373

Other interesting data:

- $\max \max_i \|v_i^S - v_i^J\| = 3.5 * 10^{-11}$.
- $\max \max_i \frac{|\lambda_i^S - \lambda_i^J|}{|\lambda_i^J|} = 5 * 10^{-14}$.
- $\min \min_i relgap_i(\sigma) = 0$.
- $\min \min_i relgap_i(\lambda) = 6.5 * 10^{-4}$.
- $5.5 * 10^2 < \kappa_2 < 9.9 * 10^{150}$.
- $\max \max(\kappa(X), \kappa(Y)) = 45.4$.
- $\max \max(\kappa(GD_G)) = 35.3$.
- Maximum relative error in eigenvalues

MATLAB = $1.5 * 10^{87}$	Stand. J-Orthog. = $4.4 * 10^{69}$